

How might the development of data mining and log analysis  
systems for the Moodle virtual learning environment  
improve computer science students' course engagement  
and encourage course designers' future engagement  
with data analysis methods for the evaluation of  
course resources?

by  
Stephen Mark Smith

A thesis submitted in partial fulfilment of the requirements of the  
University of Lincoln for the degree of  
Masters by Research in Computer Science

in the  
School of Computer Science  
College of Science

**May 2017**

## ABSTRACT

---

Virtual learning environments (VLEs) form part of modern pedagogy in education, they contain student usage data that has potential to inform and improve this pedagogy. The question this thesis explores is how might the development of data mining and log analysis systems for the Moodle virtual learning environment, improve computer science students' course engagement and encourage course designers' future engagement with data analysis methods for the evaluation of course resources? The thesis proposes that a student will complete missed work sooner if their utilisation of the VLE is automatically tracked and electronic prompts are sent when VLE activities are missed. The thesis also hypothesises that presenting a simple-to-use data mining and visualisation tool to course designers would increase their future acceptance of data mining technology for informing course design with a longer-term intent that this would improve the quality of the online learning experience ultimately improving student engagement. Exploring the two hypotheses required the development of two software artefacts, MooTwit – a tool that contacts students when they fall behind in their VLE study and MooLog – a tool that extracts and presents summative information on VLE course utilisation.

To establish if student timely engagement improved, the study used MooTwit with two groups of students over a period of 15 weeks, messaging one group only when they fell behind. Statistical analysis and comparisons were made between how quickly each group engaged with the missed items. Using MooTwit to track and contact students did influence the timeliness of their engagement with the VLE activities. Specifically, the results suggest by direct messaging a student to engage with missed material, they complete missed activities closer to required completion date. To ascertain if the acceptance of data mining for course evaluation could be improved, surveys were used before and after a demonstration of MooLog to a group of course designers. The pre-demonstration survey assessed existing planning and evaluation processes. The post-demonstration survey collected evaluations of the relevance of the information provided by MooLog and the likelihood of

the software being used to evaluate course effectiveness. The results of the study established that many designers currently do not use data analysis as a method of informing course improvement and there was evidence to suggest the MooLog demonstration significantly increased acceptance of the potential of data mining.

The findings within the thesis show that educational data-mining has the potential to improve pedagogy in VLE linked education offering opportunities to increase timely engagement and to raise course designers' acceptance of data mining to improve the validity and quality of course evaluation.

## ACKNOWLEDGEMENTS

---

*I would like to thank Dr David Cobham and Dr Kevin Jacques for their mentorship and encouragement throughout the duration of this study, your advice was truly appreciated.*

# TABLE OF CONTENTS

---

<b>1 Introduction .....</b>	<b>9</b>
<b>2 Literature review.....</b>	<b>13</b>
<b>3 Research Methods.....</b>	<b>21</b>
3.1 Student engagement system.....	22
3.1.1 Strategy of enquiry.....	22
3.1.2 Software development methods.....	24
3.1.3 Software selection .....	25
3.1.4 Communications systems .....	26
3.1.5 Moodle plug-in software development.....	27
3.1.6 Student engagement Research Design .....	40
3.1.7 Student engagement system evaluation.....	43
3.2 VLE usage analysis software .....	45
3.2.1 Strategy of enquiry.....	46
3.2.2 Software development methods.....	48
3.2.3 Evaluation .....	62
3.3 Ethics .....	63
<b>4 Data analysis.....</b>	<b>64</b>
4.1 Student engagement.....	64
4.1.1 Independent Samples t-Test.....	65
4.1.2 Comparison between accesses.....	67
4.1.3 Positive access trend comparison.....	67
4.2 VLE usage analysis software .....	70
4.2.1 Pre-development.....	70
4.2.2 Post-Development The usefulness of the information provided by the software .....	75
4.2.3 Would the tool be used if provided in normal activity? .....	80

4.2.4	Comparison of levels of acceptance between designers who previously did/did not evaluate their courses. ....	81
<b>5</b>	<b>Discussion .....</b>	<b>82</b>
5.1	Student engagement system.....	82
5.2	VLE Usage analysis software .....	87
5.2.1	Is Moodle capable of providing the tracking data in a suitable format to be further processed?.....	88
5.2.2	Can effective processing be performed on the data to provide meaningful information?.....	90
5.2.3	Can the result of the processing be displayed in an easy to interpret format for a designer to use and is the information useful to the designer? .....	91
5.2.4	VLE usage analysis software results summary .....	92
<b>6</b>	<b>Conclusions.....</b>	<b>94</b>
6.1	Conclusions drawn from MooTwit.....	96
6.2	Conclusions drawn from MooLog.....	98
6.3	Closing thoughts.....	101
<b>7</b>	<b>Reference List .....</b>	<b>102</b>
<b>8</b>	<b>Appendices.....</b>	<b>113</b>
8.1	Appendix A–Moodle engagement plug-in ERD.....	113
8.2	Appendix B-Log extraction query with sample output.....	114
8.3	Appendix C –Surveys .....	116
8.3.1	Moodle usage survey .....	116
8.3.2	Moodle Log Analyser Survey .....	118

## TABLE OF FIGURES

---

Figure 1-A typical section of a Moodle course structure .....	14
Figure 2-A Sample section of a Moodle course .....	15
Figure 3-A sample Moodle lesson module .....	15
Figure 4-Features of qualitative and quantitative research (Neill 2007) .....	22
Figure 5-Use Case Diagram Manual Student Tracking.....	28
Figure 6-Use Case Diagram Student engagement system .....	29
Figure 7-Sample XML Moodle table creation file .....	31
Figure 8-Plug-in engagement table design .....	32
Figure 9-Plug-in engagement_lecturers table .....	33
Figure 10-UML engagement class design .....	34
Figure 11-Engagement plugin display sequence diagram .....	35
Figure 12-index.php output .....	36
Figure 13-the report_engagement_cron() flowchart .....	38
Figure 14-Send message flowchart .....	39
Figure 15-MooTwit engagement plug-in .....	40
Figure 16-Twitter direct message (Failed to complete work) .....	40
Figure 17-Sample missed activity email.....	41
Figure 18-Twitter direct message (Completed all work) .....	41
Figure 19-Entity relationship diagram exported log data .....	43
Figure 20-Categories of access by students.....	44
Figure 21-Sample record of accesses per week .....	44
Figure 22-Model-View-ViewModel Principles (Microsoft 2012).....	49
Figure 23-Moodle configurable reports plug-in.....	50
Figure 24-Simplified Moodle tables and relationships used by query .....	52
Figure 25-Moodle context structure(Moodle Docs 2011) .....	54
Figure 26-Adding a date range filter to a Moodle configurable report.....	54

Figure 27-The ODS Analyse class .....	56
Figure 28-Initial user interface prototype MooLog.....	58
Figure 29-Final user interface MooLog.....	59
Figure 30-Implemented student selection window .....	60
Figure 31-WPF Databinding process(Microsoft 2010) .....	60
Figure 32-Implemented chart window .....	61
Figure 33 - Graph of student access profiles .....	67
Figure 34 - Engagement trends during study.....	68
Figure 35-Chart of responses on the importance of knowing how many of a group accessed each resource.....	75
Figure 36-Chart of responses on the importance of knowing how many total accesses there were to a Moodle resource (popularity). ....	76
Figure 37-Chart of responses on the importance of being able to identify resources that had low or zero usage. ....	77
Figure 38-Chart of responses on the importance of being able to identify resources that were used based on a student's level of achievement.....	78
Figure 39-Chart of responses on the importance of being able to find an individual student's usage. ....	79
Figure 40-Chart of responses identifying whether the analysis software would be used by the participants.....	80
Figure 41-Chart of software adoption by designers who previously did/did not evaluate their courses.....	81



# 1 INTRODUCTION

---

The use of virtual learning environments (VLEs) to augment the learning opportunities through online courses, supplementary activities and resources for learners has been identified as being utilised in a high proportion of surveyed educational institutions within the UK, (Walker, R., Voce, J. and Ahmed 2014; Ofsted & The Office for Standards in Education 2009). The adoption of such technologies has led to a wider range of tools to engage learners with topics relating to their areas of study. This increased reliance on resources that are to be used by students independently has created new challenges for both the learner and the developer; for a learner to be successful in their studies they must engage with the VLE systems and be provided with resources that allow them to learn effectively, making efficient use of the time they spend on the resources. These desirable outcomes very much link to the memorable quotes “*Good habits formed at youth make all the difference*” made by the Greek philosopher Aristotle and “*I never teach my pupils. I only attempt to provide the conditions in which they can learn*” by Albert Einstein that have inspired the aims of this study. The first aim of this study is to improve the level of engagement with Lincoln Colleges Moodle based virtual learning environment by students by instilling a good habit of learning when away from college. The second goal of the study is to improve the quality of the student experience by encouraging the future engagement of course developers with data mining systems to identify online course resources and activities that are meeting learner needs.

This intent led to the research question “How might the development of data mining and log analysis systems for the Moodle virtual learning environment, improve computer science students' course engagement and encourage course designers' future engagement with data analysis methods for the evaluation of course resources?”.

There have been many studies (Currie 2016; Liu et al. 2016; Kaur et al. 2015; Yen et al. 2015; Mohamad & Tasir 2013; Romero et al. 2013; Mazza et al. 2012; Zhang & Almeroth 2010; Dale 2007; Ortigosa & Carro 2003) into the use of data from virtual learning environments that identify the level of utilisation of the activities and resources within them. The common theme from these investigations is the success in generating the information from the data; the conclusions drawn from the studies consistently avoid establishing whether there was any positive impact as a consequence of providing the information to either the course designer/deliverer or the learners themselves. Conclusions typically make statements such as “expert teacher knowledge for learning analytics can sometimes be outperformed by knowledge derived by data-mining algorithms” (Liu et al. 2016) and “Educators with no expertise in data mining can also apply their hands in these fields.” (Kaur et al. 2015) although Kaur does evaluate the success of the accuracy of the information in identifying slow learners there is no evidence of impact. This research will focus on the impact of developing and providing two pieces of software to meet the objectives below:

1. Can student engagement with missed VLE activities be improved over time by automated log analysis and direct prompting through electronic communications systems?
2. To what extent could a system that graphically presents data mined information about course resources encourage future engagement with data analytics for informing course design?

Positive outcomes from both elements would have a direct benefit for learners and designers in the quality of the learning experience provided by the institution VLE. This study intends to establish if more effective tools will allow for better student engagement and hypothesises that:

1. Students receiving automated prompting would access missed VLE course materials in a timelier manner than unprompted students.
2. Providing a simple-to-use data mining and visualisation tool to course designers would increase their future acceptance of data mining technology for informing course design.

The investigation involves students and course developers within Lincoln College who use the Moodle VLE system for purpose of academic study and delivery of Further Education and Higher Education qualifications, with the intention improving student engagement with VLE resources by promoting the development of high quality online learning environments and study habits within the college. Should the study find evidence to support the proposed hypotheses, it is the intention to develop the software systems further to the benefit of Lincoln College and the Moodle community, offering educators additional tools to understand and support learner engagement with online study activities. It is hoped that the results will contribute to larger areas of study including the production of additional mechanisms for understanding how learners learn effectively when working independently online and how they can be encouraged to engage successfully with online learning systems that can be structured to their needs. It is also hoped that the software developments undertaken within the study will provide software that could be used in the wider educational community and a framework for future VLE data analysis developments.

The chapters that follow discuss the study in greater depth. Chapter 2 identifies the reasons for the use of virtual learning environments and how they function, the existing research on the production of tools to analyse log data and identifies the focus of the research. Chapter 3 details the research methods and software design required to conduct the research. Chapter 4 presents the summative results of the study which are evaluated in Chapter 5 with the conclusions and recommendations presented in Chapter 6.

## 2 LITERATURE REVIEW

---

Virtual learning environments and learning management systems (LMSs) are the most common systems used to deliver learning materials and activities in UK further education establishments and considered a "...pervasive technology in higher education institutions..." (Weller 2007). The pedagogical underpinning for the use of VLEs is intended to enable a transition from teacher-led delivery to a more student-centred learning experience. Their adoption has been encouraged by the government and education funded Joint Information Systems Committee (JISC) who identify digital technologies such as VLEs as "exciting opportunities to reconsider how we teach, engage with or involve learners in different ways" (JISC 2016). The use of VLEs present opportunities at the institutional, the teacher and the student level for improving and enhancing learning, hence there being a significant investment in the learning technology.

The process of learning to use VLEs effectively has presented a challenge to both the teaching staff in institutions and the students who utilise them, teachers are providing content and learning activities to students who have little experience of how to use a VLE. Teachers who have little experience of the technology as a teaching tool often miss the requirement to teach the learner how to use the VLE, consequently the students "...find it difficult to engage with technology-based tasks as they may have little prior experience." (Hughes 2005), more importantly students who do not engage with the learning may not be identified until they have fallen so far behind that they are unable to catch up with the coursework. This issue is a concern for the Computer Studies section of Lincoln College who were early adopters of VLE systems for supporting students in their learning through online systems; initially using self-devised tools, followed using the Blackboard VLE and subsequently the open source Modular Object-Oriented Dynamic Learning Environment (Moodle). Moodle is used for providing all learning materials and learning activities for the section with paper based materials being used infrequently.

Moodle allows a course designer to split a course into a structured set of activities, a typical Moodle course structure is shown in Figure 1, where a Moodle course is made up of a sequence of topics (Moodle sections) containing a number of activities (Moodle modules); some of the activities are mandatory for a student to complete the course at a minimum level and are expected to be completed in a timely manner.

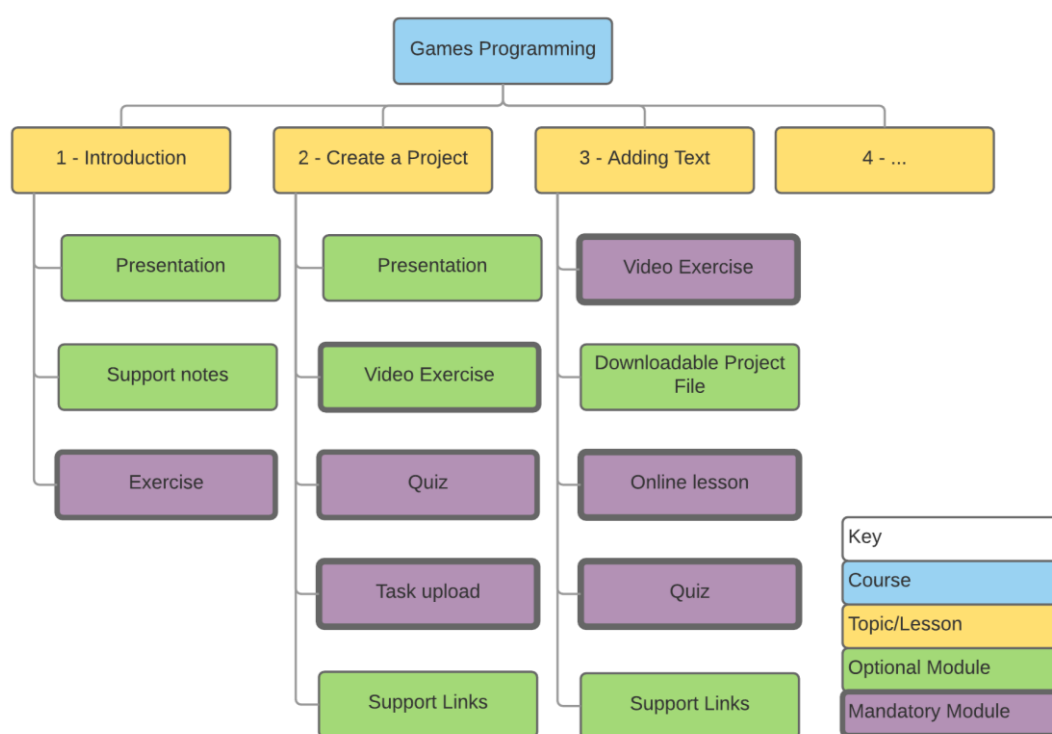


Figure 1-A typical section of a Moodle course structure

The course presents itself to the student through a web interface, that allows them to navigate the sections and engage with modules. The look and feel of the interface can be modified through the application of themes, Figure 2-A Sample section of a Moodle course shows a section of a Moodle course presented to a computing student at Lincoln College. There are four modules within the section the student should engage with:

- Introduction to XNA presentation (Pop-up URL module).
- Exercise 01 Creating an XNA project video (Page module).
- Adding text to an XNA Screen presentation (Pop-up URL module).
- Exercise 02 XNA writing to the screen (Lesson module).

### Creating an XNA Project and Getting Text On Screen

Activity	Assessment	References/Notes
Introduce XNA Game Concepts	A new XNA Game project created in Visual Studio.	Students feedback existing OOP knowledge from previous programming unit.  Exposition on the methods in an XNA game and the game loop.

#### Introduction to XNA

Activity	Assessment	References/Notes
Create an XNA game template	A new game solution created in Visual Studio  Lesson questions correctly answered	Students follow video instructions to create the project answering questions to progress through the lesson.

#### Exercise 01 Creating a XNA Project

Activity	Assessment	References/Notes
Create a sprite font and use it within an XNA game.	Completed XNA program with text on the screen.  Questions asked within the Lesson	Exposition on SpriteFonts  Students follow lesson video creating a sprite font answering the questions to progress.

#### Adding Text to an XNA Screen

#### Exercise 02 XNA Writing to the Screen

Figure 2-A Sample section of a Moodle course

‘Exercise 02 XNA Writing to the Screen’ is an example of a Moodle lesson module which would present itself to the student as a navigable lesson with quiz elements.

## Developing Computer Games

You are logged in as Admin User: Student (Return to my normal role)

English (en)

Home ► Courses ► Computing ► Unit22 ► Creating an XNA Project and Getting Text On Screen ► Exercise 02 XNA Writing to the Screen

### Exercise 02 XNA Writing to the Screen

What does the base.Initialize() command do?

☐ Called the Initialize() method of the current object.  
☐ Creates a new object  
☐ Calls the Initialize() method of the inherited class.

You have completed 25% of the lesson

25%

You are logged in as Admin User: Student (Return to my normal role)

Unit22

Figure 3-A sample Moodle lesson module

The Moodle Courses designed for the VLE are initially developed as the academic year progresses which can lead to a poorly structured first attempt and are assembled by tutors who have had no formal tuition on online course design. In subsequent years' course materials are revisited and modified generally without any analysis of how the system was used. Authors mostly refine the course through an iterative process based on observation of student activity and discussion. Ideally courses should be "Mined and interpreted for highly integrated and functional course deliveries" (Currie 2016). For a course author, there is no specific information on how the existing materials were used or accessed without them performing in-depth log retrieval and analysis relying on the designer spotting structural changes based on the complex reported information; consequently, conclusions and amendments made to courses are often based on subjective rather than objective measures and very much inclined to bias.

For course designers, this superficial reviewing of courses is not conducive to good design and is a problem area for investigation. Mohamad and Tasir concluded that, "examining how students make use of the system is one way to assess the instructional design in a formative manner and it may shed some useful insights for the educator to improve the instructional materials" (2013), supporting the intention to develop and evaluate a tool that pursues this goal. The level of VLE "...system usage is directly influenced by ease of use" (Sanchez & Hueros 2010) was identified as crucial element in their investigation into the factors that influence students to use the Moodle virtual teaching system and findings by Davis et al. in user acceptance of computer technology stating "...usefulness of the system is even more important and should not be over-looked" (1989). It is therefore necessary to identify the VLE items students found the most useful to complete a section or module and to ensure usefulness and usability of less effective items are evaluated and either refined, replaced or removed to raise the quality of the learning experience.



One of the two aims of this study is to develop software that will allow tutors to visualise how their courses are used by the students and clarify if this changes the attitudes of course designers in relation to the use of data mining for analysing the effectiveness of course elements. This software builds on previous research and development that created a Moodle block for "Visualizing and Mining Students' Usage Data" (Romero et al. 2013), that had a focus on monitoring student performance in contrast with evaluating course usage and was limited to information only held on the Moodle system. The "information visualisations" (Spence 2001) will filter how the course was accessed based on the level of attainment of students plus the popularity of each item, enabling informed course design and redevelopment decisions to be made supporting the principles of "continuous empirical evaluation in course design" (Ortigosa & Carro 2003). While good course design can have an influence on the level of engagement with the VLE technology, increased engagement could also be derived from the development of additional data analysis software focused on the learner.

Within Lincoln college's computing section, the expectation of all tutors is that students are all fully engaged with their courses on the VLE and it has been observed that in many instances this has not been the case. This situation is confirmed in another study of VLE engagement that concluded "...although some students engaged extensively with the online activities, many did not or did so less than the course team and the teachers had hoped." (Hampel & Pleines 2013). A lack of engagement with the VLE in this scenario as identified by (Morgan, 2001, cited in Maltby & Mackie 2009) can have an impact on the level of success of a student. Kuh et al. identified two key components for a student to be successful, "...the amount of time and effort the student puts into their studies..." and "...the ways the institution allocated resources and organizes learning opportunities and services to induce students to participate..." (2011). Uvalic-Trumbic and Daniel in 'A Guide to Quality in Online Learning' state that "In sum, a quality virtual learning environment is firmly based on the pedagogical needs of the course and its learners, is reliable and robust, is aligned with the technical infrastructure of the institution, and is regularly

subjected to internal evaluations, updating and improvements as needed." (2013) further reinforcing the requirement for course designers to review resources to improve quality to meet the needs of the learner encouraging engagement with their learning.

Encouraging students to participate in VLE learning activities has been attempted using the University of Wolverhampton's Online Learning Environment (WOLF); the analysis of the study recommended "...monitoring of regular habits in using WOLF, including the activities" (Dale 2007) to ensure adequate engagement. Mazza & Milani made the observation that, "Educational research literature also shows how monitoring student learning is a crucial component of high quality education." (2004), but for a tutor to check a student's engagement with a Moodle course they are required to manually navigate a HTML forms based query system and analyse the resulting report to identify what the student has and has not accessed, (Moodle Docs 2013). The process is labour intensive; regular application to a full student cohort would create significant workload for a tutor. Through the use of software automation this is an avoidable scenario, it has been identified that Moodle "...collects a large amount of data on student interactions within it, including content, assessments, and communication..." (Yen et al. 2015) that could be data mined.

Solutions have already been attempted to improve the quality of the reporting provided to tutors. The current research and development has focused on the use of log analysis to produce a "learning analytics dashboard" (Dawson 2014), charts and graphs alongside materials in Moodle (Mazza & Milani 2004) or "...provides aggregated and useful statistical reports." (Zhang & Almeroth 2010); all require the direct engagement of the designer and deliverer to interpret and action the results; typically, non-engagement of a student is resolved and identified by the deliverer, from this it expected that they will be able to identify weak spots in the course and amend the design for the next delivery session. The previous studies reporting systems have a variable level of effectiveness in terms of their ease of use, the expected technical level of

the users of the systems and a user's ability to interpret the information presented. The software developments within this study aim to provide better tools to track and prompt disengaged learners and simplify data mining of resource utilisation to improve the course quality, both having the end goal of improving course engagement.

None of the previous research has covered the use of automated stimulus control as a mechanism for improving student engagement with a VLE. Stimulus control is defined by Bloh as "...a change in operant behaviour that occurs when a particular type of stimulus is presented" , (2008). The findings of Miltenberger in the area of prompting and transfer of stimulus control indicated that through multiple prompts behaviour can be changed (2011). This study proposes the use of social networking as the stimulus based on the observation that "these applications already gained high popularity among students and are suitable to be used to engage the students..." (Mohamad & Tasir 2013) . This is supported by existing evidence of social networking being used in education, for example Junco et al. found that "...using Twitter in educationally relevant ways can increase student engagement and improve grades" (2011) and Twitter use significantly increasing student use of valid Internet sources in research as identified by Halpin, (2016). Research undertaken by Knight & Kaye (2014) into students preferred educational use of Twitter also identified that, "The three most highly ranked suggestions were the use of Twitter to provide information on practical issues, the posting of course-related reminders and the post of assignment submission deadlines". These studies provide indicators that the use of social networking could yield positive results for this study.

In summary, the goals of this research are to establish if it is possible to improve the timeliness of student engagement with VLE activities/resources and to examine if the demonstration of an evaluative tool for course designers will encourage them to engage with data mining systems to aid course refinement in the future, with the hope that better tools will lead to better student engagement and a higher quality of learner experience. This will be achieved through the development of two

software artefacts and examining the change in the study participants' behaviours and attitudes because of the utilisation and demonstration of the software. The two items to be created are:

**MooTwit** – a Moodle plug-in enabling checking student engagement with the VLE and automate the process of prompting the student regarding their lack of timely engagement. The system will perform:

- Log analysis to identify a student's level of engagement;
- Twitter and Email messaging directing students who are late accessing the learning materials on the VLE and congratulating students who complete work on time;
- Messages to course deliverers regarding lack of individual student engagement.

This development, use and the accompanying analysis of the results of the plug-ins use will provide an insight into how students engagement with the course changes and offers guidance for other researchers who would like to develop VLE plug-ins.

**MooView** – A Microsoft windows desktop application that analyses Moodle course log information showing graphical course activity information and allows a tutor to dynamically adjust the reporting criteria and see the usage patterns.

It is hoped that the use of the software by course designers will encourage them make use of data mining tools as a form of resource evaluation in the future. The accompanying research around the existing methods used by the designers and their evaluation of the information provided by MooView will identify possible metrics VLE and software developers could implement within VLEs. The provision of more effective analysis tools has the potential to motivate designers to use them to evaluate and improve the quality of course resources to improve learner engagement.

### 3 RESEARCH METHODS

---

The following sections outline the research methods selected to answer the research question objectives.

1. Can student engagement with missed VLE activities be improved over time by automated log analysis and direct prompting through electronic communications systems?
2. To what extent could a system that graphically presents data mined information about course resources encourage course designers' future engagement with data analytics for informing course design.

Given that the research requires the development of software to perform the study a significant proportion of the methodology incorporates the details of the software development methods used to produce the programs required to conduct the research. The inclusion of this information will aid other researchers in understanding the processes of collecting data from a complex VLE system such as Moodle and provide a framework that can be reused within other developments.

### 3.1 STUDENT ENGAGEMENT SYSTEM

#### 3.1.1 Strategy of enquiry

Before selecting methods of enquiry, the advantages and disadvantages of quantitative and qualitative methods were evaluated. The key features considered are identified in Figure 4-Features of qualitative and quantitative research (Neill 2007).

QUALITATIVE	QUANTITATIVE
The aim is a complete, detailed description.	The aim is to classify features, count them, and construct statistical models in an attempt to explain what is observed.
Researcher may only know roughly in advance what he/she is looking for.	Researcher knows clearly in advance what he/she is looking for.
Recommended during earlier phases of research projects.	Recommended during latter phases of research projects.
The design emerges as the study unfolds.	All aspects of the study are carefully designed before data is collected.
Researcher is the data gathering instrument.	Researcher uses tools, such as questionnaires or equipment to collect numerical data.
Data is in the form of words, pictures or objects.	Data is in the form of numbers and statistics.
Subjective - individuals' interpretation of events is important, e.g. uses participant observation, in-depth interviews etc.	Objective: seeks precise measurement & analysis of target concepts, e.g., uses surveys, questionnaires etc.
Qualitative data is more 'rich', time consuming, and less able to be generalized.	Quantitative data is more efficient, able to test hypotheses, but may miss contextual detail.
Researcher tends to become subjectively immersed in the subject matter.	Researcher tends to remain objectively separated from the subject matter.

Figure 4-Features of qualitative and quantitative research (Neill 2007)

When considering the methodology for research element 1 "Can student engagement with missed VLE activities be improved over time by automated log analysis and direct prompting through electronic communications systems?" there was a need to consider collecting of data that would allow for the evaluation of the systems success. A quantitative approach was selected based on the features identified in Figure 4-Features of qualitative and quantitative research (Neill 2007), The research makes use of data from the Moodle database logs of the students' accesses to the VLE; this aligns with Neall's quantitative criteria including, "Data is in the form of numbers and statistics" and the research is looking for specific changes in engagement meeting the criteria "Researcher knows clearly in advance what he/she is looking for".

During the planning stages of the experiment the possibility of bias was considered, while "bias can never be completely eliminated" (Lazar et al. 2010) the research methodology endeavoured to avoid bias as follows:

1. By automating the process of logging of virtual learning accesses any data collection activity is invisible to the participants, giving a lower chance of participant bias;
2. Bias due to experimenter behaviour has been reduced by using automated social media messaging generated by the engagement system ensuring all participants receive the same feedback from the system at the same time in the same format.

The use of log analysis has removed the opportunity for the learner to be aware of what they are being tracked on, obfuscation of the tracking prevents them from changing their behaviour until the system informs them they missed the work. The removal of the tutor from the process of tracking and prompting of the students limits the influence they have upon the students when they miss the work, lowering the impact on how quickly the student might complete missed work.

### 3.1.2 Software development methods

The system was developed as a proof of concept that a plug-in linked to Moodle and Twitter could be used to track and improve timely engagement with important course resources on the VLE. As a proof of concept, the plug-in is scalable in line with the database and server systems that support the VLE, its' restriction is the extent that social media limits the use of automated direct messaging. The development process used the spiral model "used commonly in developing systems" (Mohammed et al. 2010) to allow for refinement of the system through a set of controlled iterations, Adaptive Software Development (ASD) system was considered but its features are designed "for the development of large and complex system" (Khan et al. 2014) and therefore it was deemed to be inappropriate in this case.

The spiral model allowed for better planning and fixing of milestones keeping the development on track, the risk driven approach to the models steps allowed for flexibility around where the most effort needed to be focused at each stage. The development of the software went through three distinct iterations:

1. Data extraction – the first iteration (prototype 1) developed the underlying system functionality of a Moodle plugin and investigated Moodle API call output. Within the iteration integration testing took place for the plug-in and the analysis of the data-structures produced by the poorly documented Moodle system calls. The result of this stage was a plug-in that could be installed into any Moodle system through the VLE's standard mechanisms. The plug-in extracted the data about any course it was enabled in and created the necessary tables in the Moodle database as part of its installation process.
2. Tutor interface and tracking processes – the second iteration (prototype 2) developed and trialled the tutor interface within the plug-in, it provided a formatted list of resources alongside selectors that allowed a resource to be selected and a tracking date to be



assigned to it. User acceptance testing was undertaken at this stage to ensure the interface performed effectively. The result of this stage was a plug-in that allowed tutors to see a summary of the resources in their course and then select the resources to be tracked by date at a section or individual level. Selections made by tutors were stored internally in the Moodle database and linked to the code that tracked student engagement.

3. External communications – the final iteration (operational prototype) established the links between the plug-in and the twitter API and messaging through the email systems. Tests were undertaken to ensure selections made by tutors resulted in the correct messages being sent and records being updated in the data tables. This stage completed the development process allowing the study to continue.

The ability to remove bugs prior to the next phase and the quantity of the code per iteration being reduced to manageable levels allowed effective tracking and the internal processing, user interaction and output in the development key milestones to be met. Restricting the user interaction element to one iteration reduced delays in development in the other phases and reduced the debugging workload in user interaction stage. This suggests the model is an effective approach to the development, allowing the elements of the software to be produced independently and testing to take place at each phase.

### **3.1.3 Software selection**

Development was limited to the Moodle VLE based on the skill set of the users involved in the study and the system being used operationally within Lincoln College. The main development language for the engagement system was PHP: Hypertext Pre-processor (PHP) a HTML-embedded scripting language. The rationale for this was based on it allowing “web developers to write dynamically generated pages quickly” (The PHP Group 2016) and its compatibility with the Moodle VLE which is written in PHP. This aided the development through the ability to call the

existing Moodle log API, (Moodle Docs, 2013). As part of the development required a database for data storage and manipulation, Microsoft SQL server and PostgreSQL were considered alongside the MySQL database engine that was ultimately selected primarily because of its existing use in the VLE installation. Although MySQL's suitability was confirmed by Davies & Fisk who identified that it could "compete fully with proprietary databases such as Oracle and Microsoft SQL Server" (2006), PostgreSQL was considered based on its "performance appeared to be about two times better than that of MySQL" (Suh & Snodgrass 2016) in a multi-processor system, but given that the database system is externally hosted it could not be guaranteed that this would make a difference and that the hosting service did not natively support Microsoft or PostgreSQL. To attempt to ensure portability of the resulting software where possible, data will be extracted from the database using the Moodle database API that abstracts the database functionality.

#### **3.1.4 Communications systems**

To enable communication of the level of engagement to a student requires a link to a current social networking messaging infrastructure for direct messaging; Facebook was initially considered as the system for messaging but it has a number of restrictions relating to the sending of messages that prevents automation, Facebooks' suggested alternative of using requests are not viable given that "Requests are only available for games on Facebook.com or iOS and Android apps." (Facebook 2013). Investigation into Instagram as direct messaging platform revealed there is no API support for this facility in Instagram (Instagram 2014). Twitter was finally selected as the most appropriate system for the experimental direct messaging. A supporting factor for its choice was that Lincoln College use Twitter with students on a regular basis and its API is open to developers to use freely; although there is a restriction of "250 direct messages per day" (Twitter 2014) per user, which does limit the scalability of the end product. The quantity of messages to be sent on any one day within the study would be no greater than 100, the limit of 250 messages will not be exceeded confirming it is suitability for proof of concept for the messaging system to improve engagement. The hosting service selected for this study

offers a “built-in PHP mail function that will allow you to send an e-mail directly from a PHP script” (1&1 Internet Inc. 2015) that is to be used for delivering a summary of course elements missed each week to both the student and the course coordinator.

### **3.1.5 Moodle plug-in software development**

To integrate the required functionality into the Moodle VLE a Moodle plug-in was selected as the target for development in preference to the construction of a discreet website to perform the task. The Moodle VLE has the capability to extend itself through the use of a plug-in based system which is “the easiest and most maintainable way to add new functionality to Moodle...” (Moodle Docs 2016b). The use of this system also ensured that access to the data relating to the student was afforded the same level of security as the VLE, conforming with the Technical element of the learning analytics “DELICATE checklist” (Drachsler & Greller 2016) .

The plug-in type selected was based on the Moodle report plug-in type designed to provide “...useful views of data in a Moodle site” (Moodle Docs 2016b). The process of developing the plug-in requires the construction of a number of server side scripts written in the PHP scripting language to enable the installation, upgrade and running of the plug-in. The software design of the plug-in requires the use of both procedural design for the HTML/PHP page display and automated CRON (Command Run ON) functions; in-line with the Moodle design principles, object oriented development was used for the user interface elements. The plug-in also requires the design and implementation of tables in the Moodle database to allow data persistence and constructing SQL (Structured Query Language) queries to examine the existing Moodle database tables to retrieve course and student access data.

Software design toolsets were considered prior to the development of the plug-in; individually Booch (Booch 1999), Rumbaugh and Jacobsen (Jacobsen et al. 1999) methodologies were reviewed. Further research identified Unified Modelling Language “... UML has emerged as the

standard notation for object-oriented modelling and design". (Sommerville 2015) which is supported by Curtis & Cobham who identified UML as the "...de facto standard for object-oriented development" (2008). Where possible within the scope of the development UML principles have been applied.

### ***Initial use case for existing manual tracking process***

This initial development required the analysis of the manual tasks tutors would undertake to track students' activity via Moodle, as shown in Figure 5.

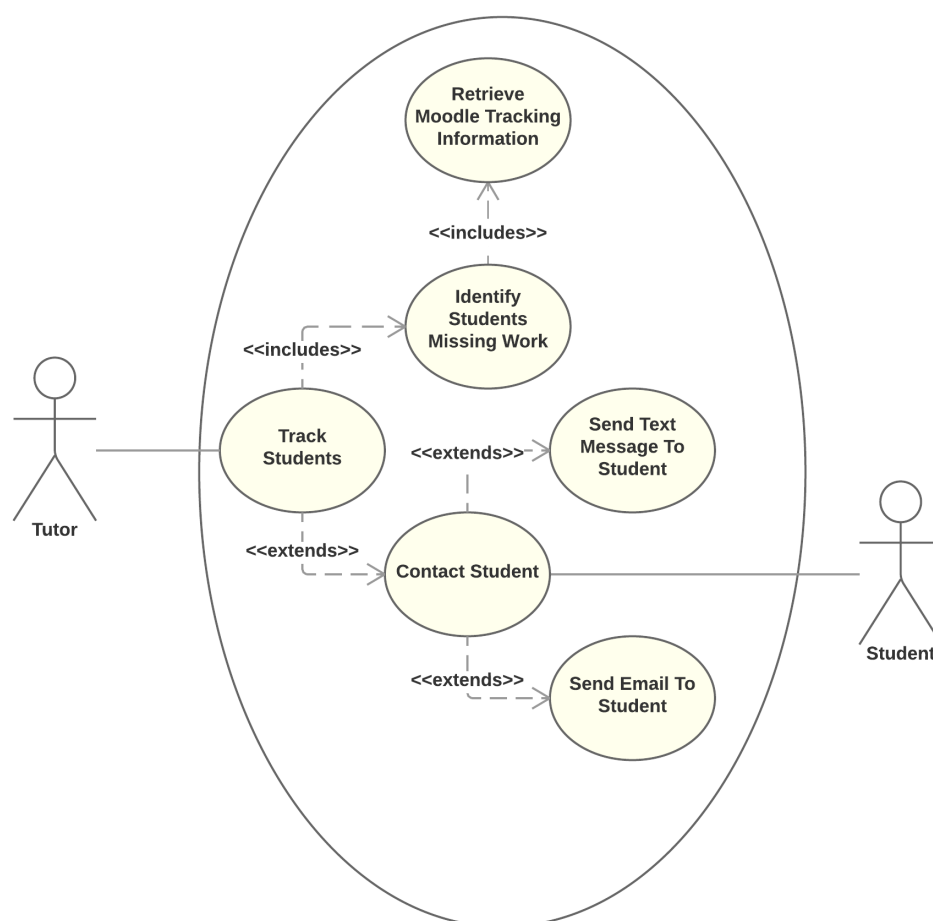


Figure 5-Use Case Diagram Manual Student Tracking

The simple analysis above formed the basis for the design of the UML use case for the enhanced automated system in Figure 6 shown on page 29.

### Proposed automated use case based on initial findings

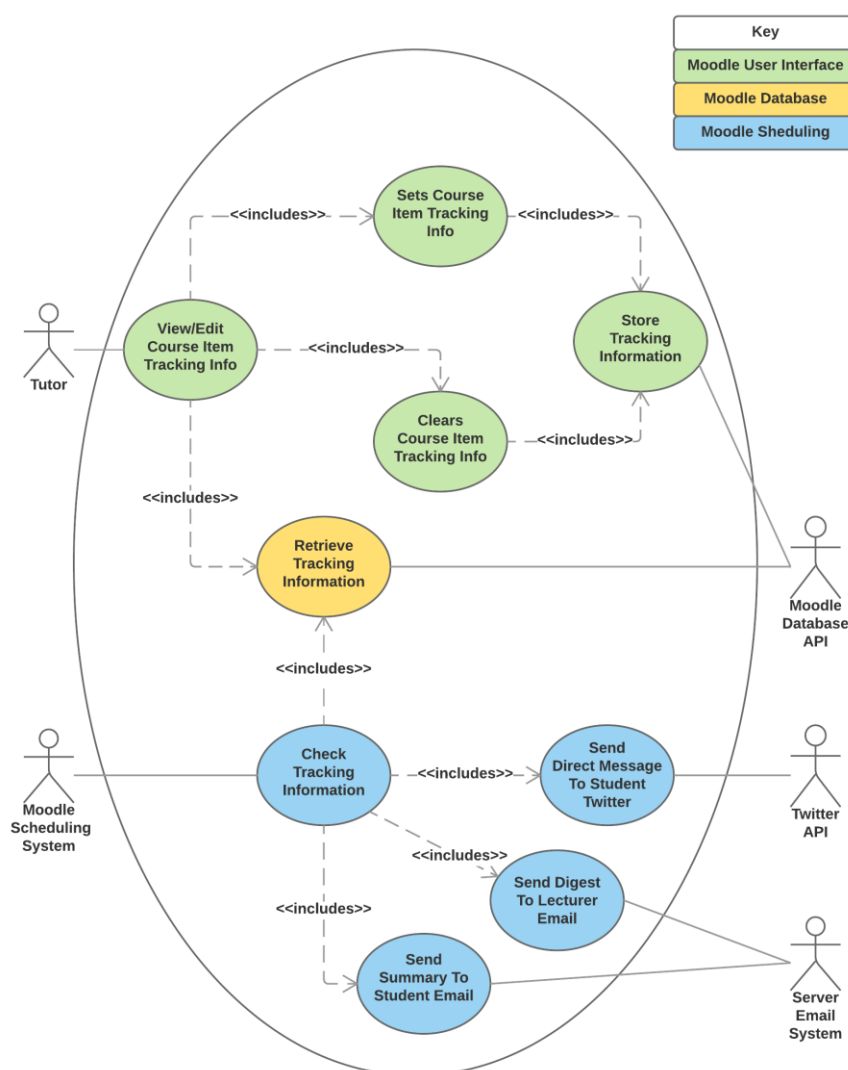


Figure 6-Use Case Diagram Student engagement system

The final use case identified three key areas of development for the plug-in to work.

1. User interaction via the plug-in HTML/PHP interface and the Moodle forms API.
2. Automating student tracking through the plug-in Command Run ON (CRON) functions.
3. API interaction with external systems (database/twitter/email).

To achieve this required the writing of PHP code within a standard Moodle plug-in framework which is "The easiest and most maintainable way to add new functionality to Moodle" (Moodle Docs 2016a). Moodle plug-

ins follow a standard file structure layout and naming convention allowing integration into the VLE system without modification of the main code. Table 1 identifies folder structure and the purpose of the files created for the engagement plug-in developed for the experiment.

<b>engagement</b>	<i>Plug-in folder</i>
config.php	<i>Plug-in developer configuration file</i>
index.php	<i>Template page to display in Moodle when the plug-in is selected by user</i>
lib.php	<i>PHP functions that link to Moodle system calls e.g. &lt;plug-in&gt;_cron()</i>
locallib.php	<i>Main custom plug-in code</i>
styles.css	<i>Custom CSS styles for plug-in</i>
version.php	<i>Plug-in version and Moodle compatibility version</i>
<b>classes</b>	<i>Contains any auto load class files for the plug-in</i>
<b>event</b>	<i>Contains event response code</i>
content_viewed.php	<i>Code called if plug-in is viewed</i>
<b>db</b>	<i>Contains database related files</i>
access.php	<i>Sets the access permissions for the plug-in</i>
install.xml	<i>Contains an xml definition of the plug-in database schema created by Moodle Table designer</i>
upgrade.php	<i>Contains code to upgrade a plug-ins tables based on the plug-in version number in version.php</i>
<b>lang</b>	<i>Localisation folder containing language specific files</i>
<b>en</b>	<i>English folder</i>
report_engagement.php	<i>English localisation files</i>
<b>twitteroauth</b>	<i>Separate library files (none Moodle)</i>
OAuth.php	<i>Abraham Williams oauth Twitter API library</i>
twitteroauth.php	<i>Configuration for library</i>

Table 1-Moodle engagement plug-in file and folder structure

### **Plug-in database development**

The Moodle VLE with a basic set of plugins installed has an existing database of approximately 240 tables to store data relevant to the operation of itself and the plug-ins incorporated into the system. When a plug-in is installed into Moodle an XML (eXtensible Markup Language) install.xml file in the plug-in folder is used to create the database tables within the Moodle system as shown in Figure 7-Sample XML Moodle table creation file. This system provides an abstracted table model allowing table construction in any underlying database system supported by the VLE.

```
<?xml version="1.0" encoding="UTF-8" ?>

<XMLDB PATH="report/engagement/db" VERSION="20141231"
  COMMENT="XMLDB file for Moodle report/engagement"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../../lib/xmldb/xmldb.xsd">

  <TABLES>

    <TABLE NAME="report_engagement"
      COMMENT="Record of the tracking used within the course">
      <FIELDS>
        <FIELD NAME="id" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="true"/>
        <FIELD NAME="timemodified" TYPE="int" LENGTH="10" NOTNULL="true"
          SEQUENCE="false" COMMENT="The timestamp of when modified"/>
        <FIELD NAME="courseid" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="false"
          COMMENT="Moodles internal course id number"/>
        <FIELD NAME="moduleid" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="false"
          COMMENT="Moodles internal id for a module"/>
        <FIELD NAME="groupid" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0"
          SEQUENCE="false" COMMENT="The group to be monitored 0 means no
          grouping"/>
        <FIELD NAME="completeby" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="false"
          COMMENT="a timestamp of the time the module is to be completed by"/>
      </FIELDS>
      <KEYS>
        <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
      </KEYS>
    </TABLE>

  </TABLES>

</XMLDB>
```

Figure 7-Sample XML Moodle table creation file

For the engagement plug-in, a simple 2 table design was created to allow for storage of tracking information linked to the existing Moodle tables via the VLEs log table, full details of the relationship can be seen in the simplified Moodle ERD in Appendix A–Moodle engagement plug-in ERD.

**The engagement table** – Provides the plug-in with the ability to record each tracked module in a course, the data stored in a row is identified in Figure 8-Plug-in engagement table design. Data is inserted into the engagement table when a lecturer selects items to be tracked in the engagement plugin page of a Moodle course forming part of the 'View/Edit course item tracking info' use case shown in Figure 6-Use Case Diagram Student engagement system.

A record of modules to be tracked within a course

report_engagement		
PK	id	bigint
	timemodified	bigint
	courseid	bigint
	moduleid	bigint
	groupid	bigint
	completeby	bigint

id

Unique id for each row in the table

timemodified

A timestamp to record the last access

courseid

The Moodle course id the tracked module belongs to.

moduleid

The Moodle module id to be tracked.

groupid

The group to track (un-implemented)

completeby

The date the tracked item should be completed by the student.

Figure 8-Plug-in engagement table design



**The engagement\_lecturers table** – Holds the details of the lecturers who are subscribed to the digest emails used to communicate the student tracking information on a course. The table is populated by a lecturer checking the subscribe to digest checkbox in the engagement plugin page of a Moodle course. The information held in the table is used to send the digest email to the lecturer as part of the automated tracking processes of the plug-in that is part of the 'Send digest email to tutor' use case shown in Figure 6-Use Case Diagram Student engagement system.

Contains details of lecturers to be contacted regarding the courses engagement

report_engagement_lecturers		
PK	id	bigint
	timemodified	bigint
	courseid	bigint
	userid	bigint
	fullname	char(50)
	email	char(255)

---

<b>id</b>	<i>Unique id for each row in the table</i>
-----------	--

---

<b>timemodified</b>	<i>A timestamp to record the last access</i>
---------------------	--

---

<b>courseid</b>	<i>The Moodle course id of the course the lecturer wishes to track.</i>
-----------------	---

---

<b>userid</b>	<i>The user id of the lecturer</i>
---------------	------------------------------------

---

<b>fullname</b>	<i>Full name of the lecturer to be used in the email</i>
-----------------	--

---

<b>email</b>	<i>The reply email address of the lecturer</i>
--------------	--

Figure 9-Plug-in engagement\_lecturers table

There are no specific relationships between the two tables within the scope of the plug-in other than the course id, any other relationships between the them are derived from the existing Moodle database schema in Appendix A–Moodle engagement plug-in ERD.

### Plug-in user interface design

The main user functionality of the plug-in is provided by the engagement class in the 'locallib.php' file, the class inherits some of its' functionality from the Moodle forms API built into the VLE as shown in Figure 10-UML engagement class design. It is possible to create the plug-in without the API, however the advantage of using the API is that it provides "...improved accessibility and security." (Moodle Docs 2016a) and is visually consistent with the Moodle themes selected by the user.

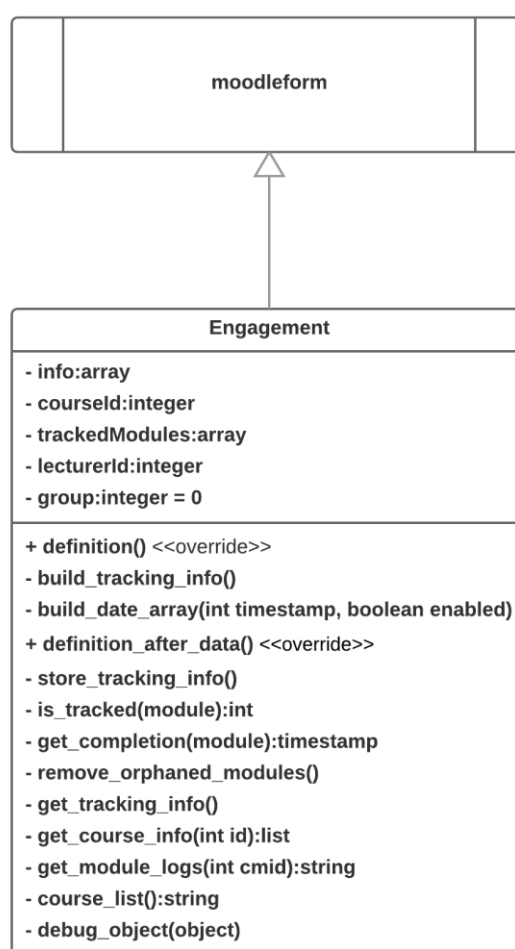


Figure 10-UML engagement class design

The Moodle plugin user interface is provided by the 'index.php' page containing procedural PHP code combined with HTML to generate the page and instantiate an object from the engagement class as shown in Figure 11-Engagement plugin display sequence diagram.

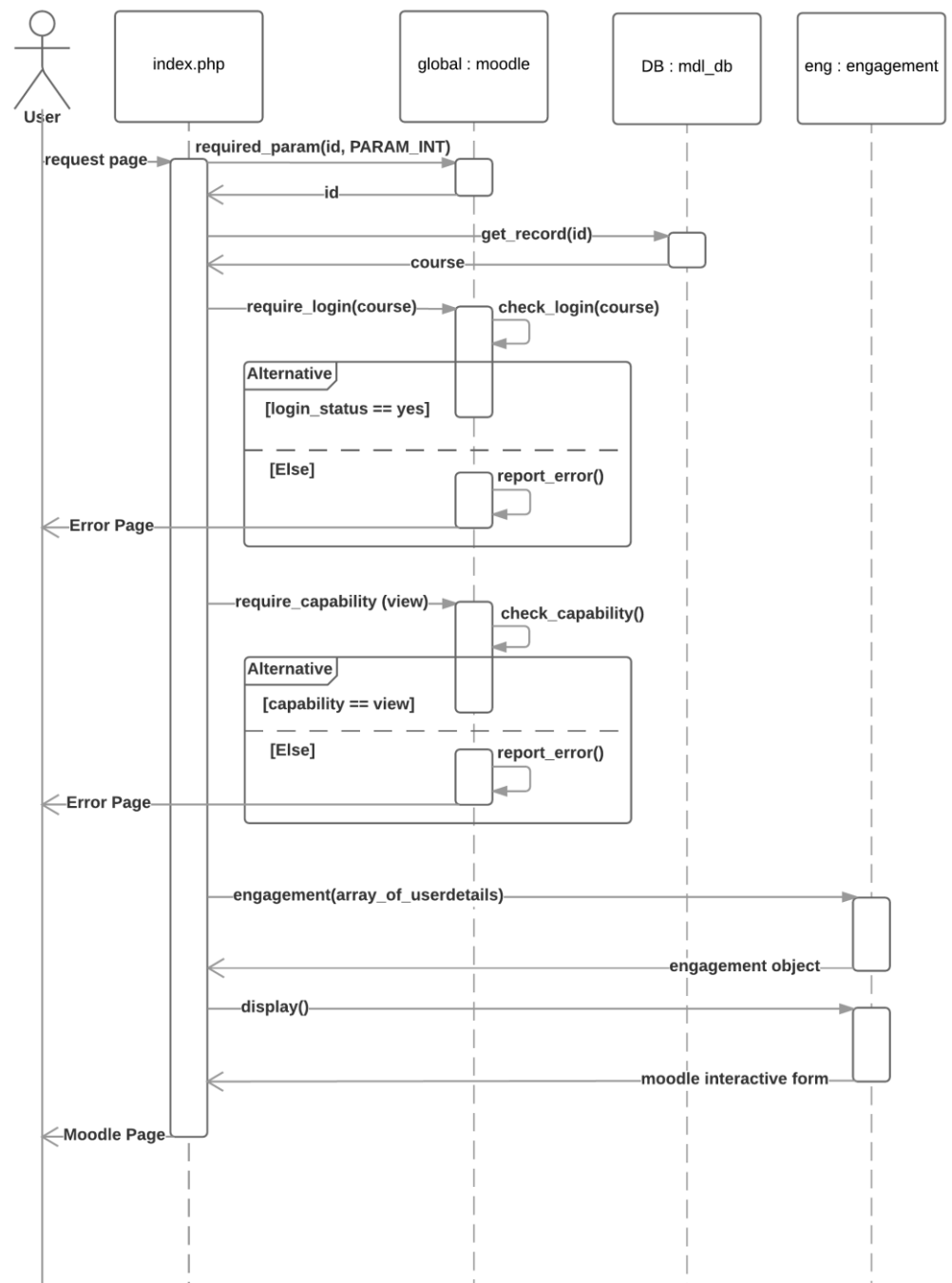


Figure 11-Engagement plugin display sequence diagram

Upon display of the engagement plug-in Moodle page, the user interacts with the form using the browser.

Information altered by the user is stored in the plug-in database tables by the engagement object reading values submitted from the form displayed on the plug-in page as shown in Figure 12-index.php output.

### Engagement report

Subscribe to the digest report ☐ Emails will be sent to moodle@

▼ Section : 000 : Unit Information

Track all elements on the same date

Section Tracking Date

News forum

Unit Specification Developing Computer Games

Assignment 1 - Computer Gaming U22 [P1, P2, D1]

Assignment 2 - Computer Gaming U22 [P3, M1, D2]

Assignment 3 - Computer Gaming U22 [P4, P5, P6, P7, M2, M3, M4]

Assignment 3 - Visual Studio Project

Video guide to setting up word to write a report

Click this link at the start of your unit and follow frustr8dlec on twitter.

XNA Coding Tips

A Guide to using my classes in your XNA project (ScreenText, AbstractSprite, Delay, Collision)

A great source of 2D Graphics

Retro Sound Generator

20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
31 ▼	January ▼	2015 ▼		<input checked="" type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>
20 ▼	March ▼	2016 ▼		<input type="checkbox"/>

Figure 12-index.php output

### Plug-in CRON function development

Within a Moodle plug-in there is a facility to execute a PHP function on a regular basis. The frequency of execution is controlled through the 'version.php' file with the line `$plugin->cron = <delay rate in seconds>;` enabling execution of a function '`<plugin type>_<plugin name>_cron()`' in the plugins 'lib.php' file.

CRON functionality was the only method for achieving this during the experiment restricting this part of the development to procedural code. Moodle is an ongoing development and from version 2.7 Moodle provides an OO solution in a Task API (Moodle Docs 2016c), that is recommended to be used in future developments.

The CRON function provides the core automation for the plug-in performing the following tasks:

1. Retrieving monitored tasks from the database.
2. Checking student completion of monitored tasks.
3. Contacting students to inform them of incomplete tasks.
4. Creating and emailing to registered staff a summary of student progress.

Figure 13-the `report_engagement_cron()` flowchart describes the processing undertaken to achieve tasks one and two, Figure 14-Send message flowchart shows the detail of the processing for tasks 3 and 4.

The retrieval of log, tracking and student data made use of Structured Query Language (SQL) queries that linked the plug-in tables to the internal tables in the Moodle database. Queries were sent using the Moodle data manipulation API that provides “a high level of abstraction and guarantee that your database manipulation will work against different RDBMSes” (Moodle Docs 2015a). This eliminated any custom database access that might break the function on different Moodle installations.

Connection to the Twitter API to send direct messages to the students used Twitters' Representational State Transfer (REST) interface, a “foundation for the modern Web architecture” (Fielding 2000). The messaging code needed to authenticate with Twitter using Open Authentication(OAuth) “an open standard for authentication, adopted by Twitter to provide access to protected information” (Kumar et al. 2013). To facilitate authentication TwitterOAuth “The most popular PHP library for use with the Twitter OAuth REST API” (Williams 2016) was incorporated into the plug-in.

Flowchart of the report\_engagement\_cron() function in 'lib.php'

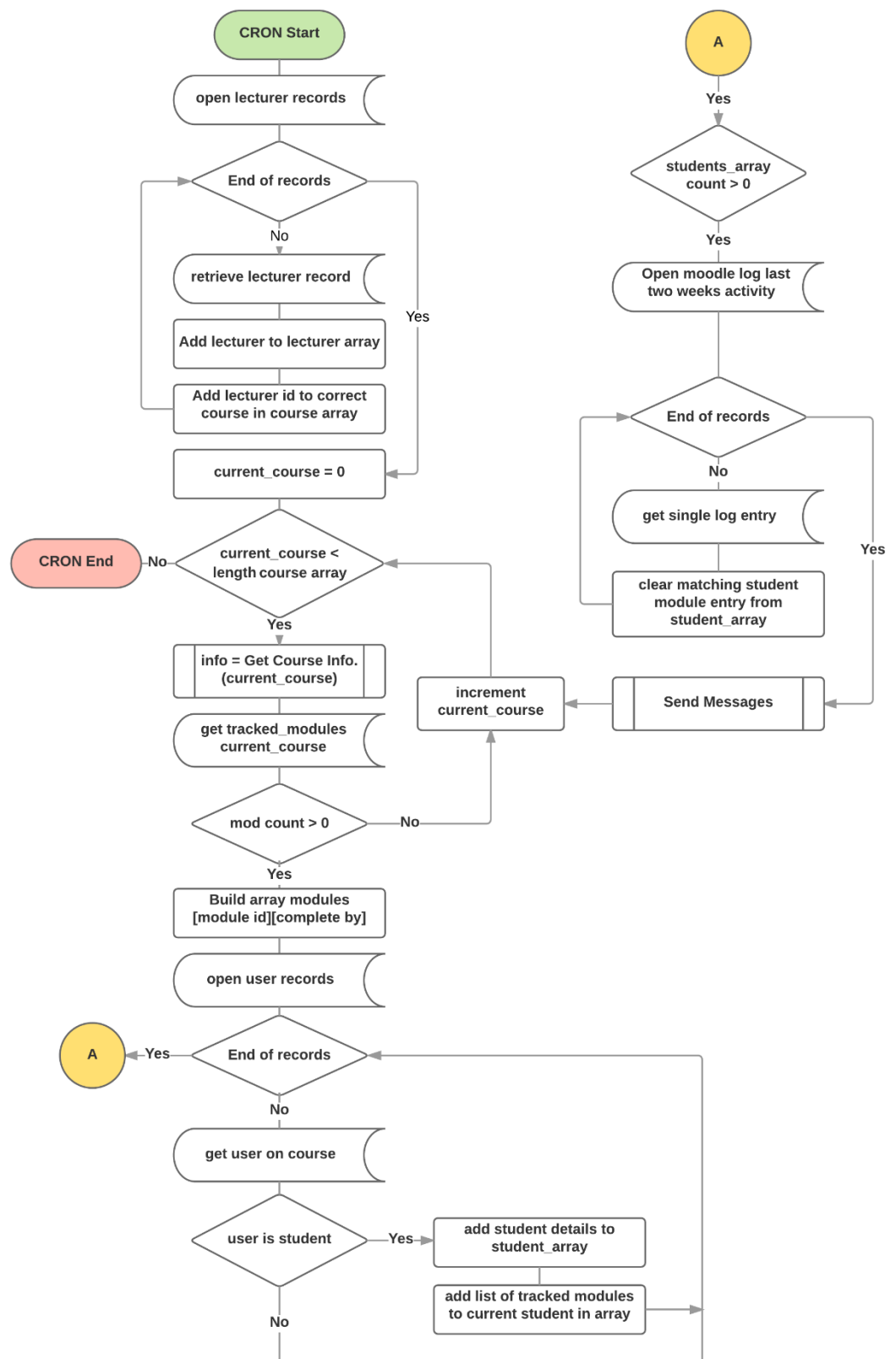


Figure 13-the report\_engagement\_cron() flowchart

## Flowchart of the Send Messages process

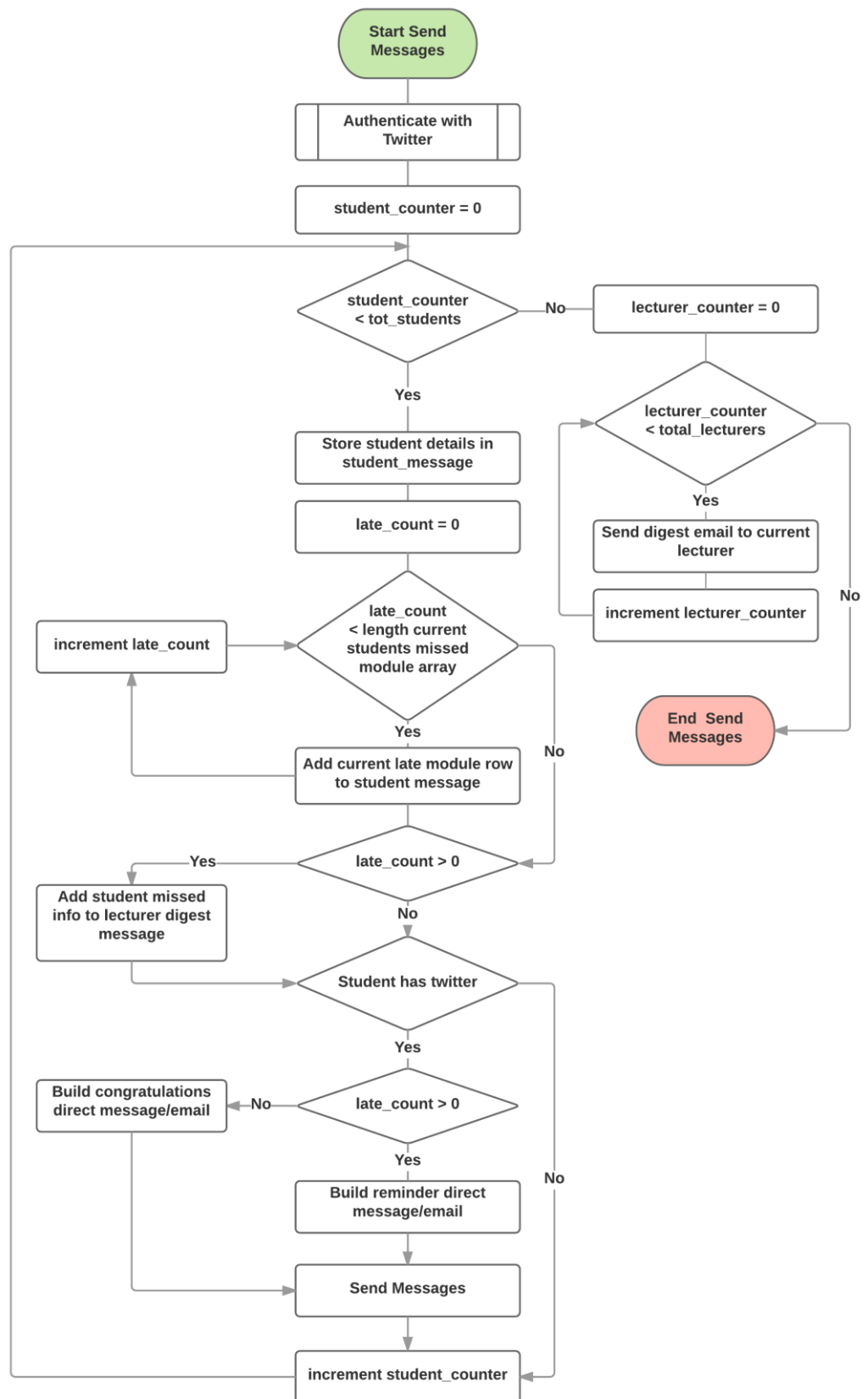


Figure 14-Send message flowchart

### 3.1.6 Student engagement Research Design

To establish if student engagement can be improved by log analysis and direct message prompting the Moodle plug-in shown in Figure 15-MooTwit engagement plug-in was developed to be used by the course subject tutor. This allowed the tutor to select either full sections of a course to be completed by a specified date or individual activities. Students who failed to complete an activity by the deadline were digitally messaged once per week for two weeks to prompt them to engage with it.

#### Engagement report

Subscribe to the digest report ☐ Emails will be sent to moodle@

▼ Section : 000 : Unit Information

Track all elements on the same date ☐

Section Tracking Date

News forum	20	March	2016	<input type="checkbox"/>
Unit Specification Developing Computer Games	20	March	2016	<input type="checkbox"/>
Assignment 1 - Computer Gaming U22 [P1, P2, D1]	20	March	2016	<input type="checkbox"/>
Assignment 2 - Computer Gaming U22 [P3, M1, D2]	31	January	2015	<input checked="" type="checkbox"/>
Assignment 3 - Computer Gaming U22 [P4, P5, P6, P7, M2, M3, M4]	20	March	2016	<input type="checkbox"/>
Assignment 3 - Visual Studio Project	20	March	2016	<input type="checkbox"/>
Video guide to setting up word to write a report	20	March	2016	<input type="checkbox"/>
Click this link at the start of your unit and follow frustr8dlec on twitter.	20	March	2016	<input type="checkbox"/>
XNA Coding Tips	20	March	2016	<input type="checkbox"/>
A Guide to using my classes in your XNA project (ScreenText, AbstractSprite, Delay, Collision)	20	March	2016	<input type="checkbox"/>
A great source of 2D Graphics	20	March	2016	<input type="checkbox"/>
Retro Sound Generator	20	March	2016	<input type="checkbox"/>

Figure 15-MooTwit engagement plug-in

The information required to identify success or failure in completing activities was gathered from the VLE log files by the plug-in. Students were then prompted with an automated Twitter direct message identifying they had missed work in the last two weeks Figure 16-Twitter direct message (Failed to complete work), backed up with a more detailed email identifying the areas of study missed shown in Figure 17-Sample missed activity email.

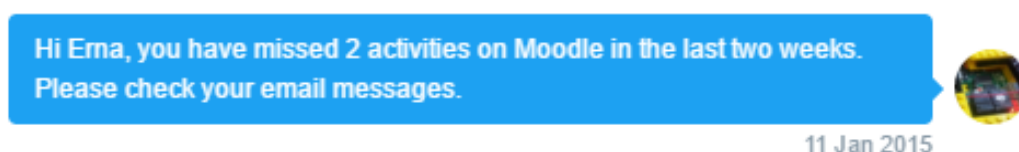


Figure 16-Twitter direct message (Failed to complete work)



To: [Student Email]  
 From: [Lecturer Email]

Subject: Engagement Report

Hi Erna,

You seem to have missed some work on Moodle in the last two weeks. To catch up you need to complete the following:

Unit 22 Developing Computer Games in section 0 Unit Information, Assignment 1 - Computer Gaming U22 [P1, P2, D1] should have been completed by: 31 January 2017

Unit 22 Developing Computer Games in section 2 Assignment Computer Gaming, Survey Question - I understand what I am required to do to complete assignment 1 should have been completed by: 31 January 2017

Regards

[Lecturer Name]

Figure 17-Sample missed activity email

If the student had completed all the work a message congratulating them was sent instead Figure 18-Twitter direct message (Completed all work).

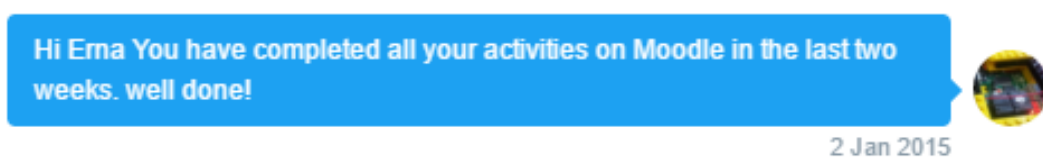


Figure 18-Twitter direct message (Completed all work)

The study was performed using two separate classes of students, both were second year groups of a level 3 BTEC Extended Diploma in IT course who had joined with the same entry criteria and studied the same subjects with the same tutors prior to the study commencing. One class formed the control group, the other the experimental group, both were informed of the study and consented to take part but were unaware of whether they were the control group or experimental group, neither group was aware of the others participation. To reduce the possibility that one group might influence the other, communications about the study between the two groups was limited by timetabling their courses on different days of the week.

The study took place over 15 weeks with 14 students in the control group and 15 in the experimental group. The control group originally had 15 members but one member was taken ill in the first week of the study and was unable to contribute.

In the first week of the study both groups were prompted to provide their Twitter name. For the study the control group would be opted out of the engagement system and their Twitter and email accounts would not receive any communications.

Over the course of the 15 weeks 57 monitored deadlines were set across two units.

- Developing Computer Games – delivered by the author
- Website Development – delivered by a subject specialist

### 3.1.7 Student engagement system evaluation

The evaluation of the success of the system was defined by the hypothesis that the experimental group will access missed material sooner than the control group as a consequence of being prompted by the system. The null hypothesis is that the experimental group will not access the learning activities any earlier than the control group after being prompted. Evaluation of the engagement system involved the comparison of log data from the control group with the experimental group. To complete this process, the data from the log table in Moodle was exported to a single table in Microsoft Access which was then split into relational tables as shown in Figure 19-Entity relationship diagram exported log data.

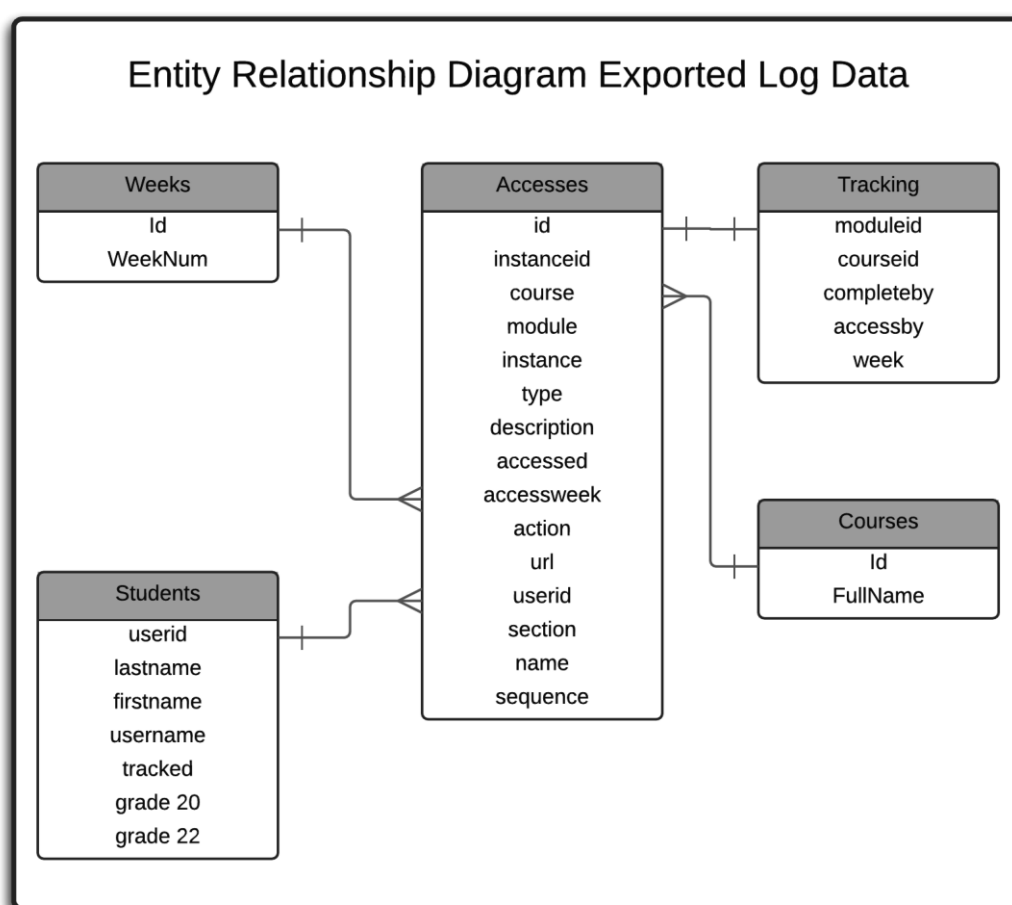


Figure 19-Entity relationship diagram exported log data

To enable the tracking of the student behaviour in a consistent manner the tracking dates were resolved to weeks from the start of the course; access measurements were then based around the week of access to ease the classification of timely access to the learning activities. Access

results were based on the first access a student made to a learning activity and were categorised as per Figure 20-Categories of access by students.

Category	Condition
Positive accesses	Took place by the deadline or within the 2-week window of prompting (student is on track)
Late access	Took place after the 2-week window of prompting (student has fallen behind)
Missed access	The student failed to access the activity within the limits of the course.

Figure 20-Categories of access by students

For each group the accesses were tabulated as shown in Figure 21-Sample record of accesses per week, the total number of weekly accesses after the expected week of access being recorded.

Activity Description	Access By	Positive Accesses Total	Negative Accesses Total	Total Accesses	Max Possible Access	Positive Access Counts			Negative Access Counts			
						On Time/Before Prompt	After First Prompt	After Second Prompt	Over 2 Weeks Late	Over 3 Weeks Late	Over 4 Weeks Late	Over 5 Weeks Late
Upload your controller information here.	2	13	0	13	14	2	10	1				
Example details of how to describe a game technology	3	8	3	11	14		7	1		2	1	

Figure 21-Sample record of accesses per week

Specifically, the evaluation will be looking at the changes in student access to the VLE on both a group access level measuring the changes before and after the system; and further inspection at regular periods during the research to gain further insights into variations in engagement over time. Conclusions drawn from the evaluation will identify the successes and/or failures of the system in improving engagement and provide a model for others to replicate with other student groups.

### 3.2 VLE USAGE ANALYSIS SOFTWARE

The VLE usage analysis software's role is to automate the analysis of the access logs relating to a specific course and communicate to the designer a breakdown of the course utilisation in a format that is easy to understand. Research by Romero, Ventura and García (2008) and Kaur, Singh and Josan (2015) used existing data mining tools WEKA (Waikato Environment for Knowledge Analysis) machine learning algorithms and KEEL (Knowledge Extraction based on Evolutionary Learning)" ...a software tool to assess evolutionary algorithms for Data Mining..." (Liu et al. 2016) to extract and analyse data for checking student performance. The weakness in their methodology was that the intended user had to recreate the whole processing system, configuring and setting up WEKA and KEEL to achieve a result. The solution proposed for this system is a bespoke software artefact aimed at ease of use for the Moodle course developer, eliminating the need for generic tools and their configuration. The expectation is that a better more user-friendly tool will improve designers' engagement with data mining and its resulting information to improve the quality of their learning resources.

Visual Studio 2015 is to be used as the development environment because of its full support of the C# language and its built-in data source exploration tools. The use of Mono (an open source implementation of C# and Microsoft .Net framework) was considered as it allows "developers to easily create cross platform applications part of the .NET Foundation" (Mono Project 2016). The option of using Mono was rejected based on the user base for the system being solely windows users, and reducing the overall workload of complying with platform specific problems such as "...non-platform specific path delimiters and line ending sequences..." and "...where to store data files an application might need..." (Sekelsenmat 2015) during development, therefore there wasn't any perceived advantage to developing a cross-platform application.

### 3.2.1 Strategy of enquiry

For the VLE usage analysis software, the aim of the research was to investigate if it is possible to present course log summary to allow the course designer to evaluate the utilisation of individual elements of a course and from this identify what extent the system would encourage future engagement with data analytics for informing course design. This was broken down into following objectives:

***Objective A - Is Moodle capable of providing the tracking data in a suitable format to be further processed?***

The initial investigation was to identify if the software would have a tangible reason for being developed and establish that Moodle can provide the tracking data in a suitable format to be further processed. A significant risk to this objective would be to assume that there was a tangible reason to develop software. To remove this uncertainty Boehm et al. when discussing the spiral model of development suggests questionnaires are “a cost-effective strategy for resolving the sources of risk” (1987). This is supported by the Center for Evidence-Based Management (2014) and Petticrew & Roberts (2003) who identify that a cross-sectional survey is an appropriate method of identifying how often a method is applied (in this case course evaluation). The process of identifying whether Moodle can provide data suitable for processing can be achieved with a literature search identifying existing research undertaken in this area and establishing whether the research had been successful and that the data contained in the Moodle VLE can be exported in a data format appropriate for processing to take place offline.

***Objective B - Can effective processing be performed on the data to provide meaningful information?***

Based on findings from the literature review confirming that the data is available from Moodle a software development process will be undertaken to create an artefact that will collate the output file from Moodle into data-structures suitable for use within a Microsoft windows application.

***Objective C - Can the result of the processing be displayed in an easy to interpret format for a designer to use?***

The resulting data structures need to be presented in a format that is meaningful. This will be achieved by reviewing existing data presentation principles and applying them in the form of a paper prototype of the system. Paper prototyping was selected to fit with the time limitations of the study; it enables quick development of a useable system and “low-fidelity prototypes appear to be as effective as high-fidelity prototypes at detecting many types of usability issues” (Affairs 2013) when developing the systems user interface (UI).

***Objective D - Is the information provided by the software useful to the designer?***

The resulting application will be presented to Moodle course designers in the form of a video presentation establishing the features of the system. A survey is to be used to establish quantitative feedback on the suitability of the system within a course development environment. The choice of a survey was based on it being “more likely than some other approaches to obtain data based on a representative sample, and can therefore be generalizable to a population.” (Kelley et al. 2003).

### 3.2.2 Software development methods

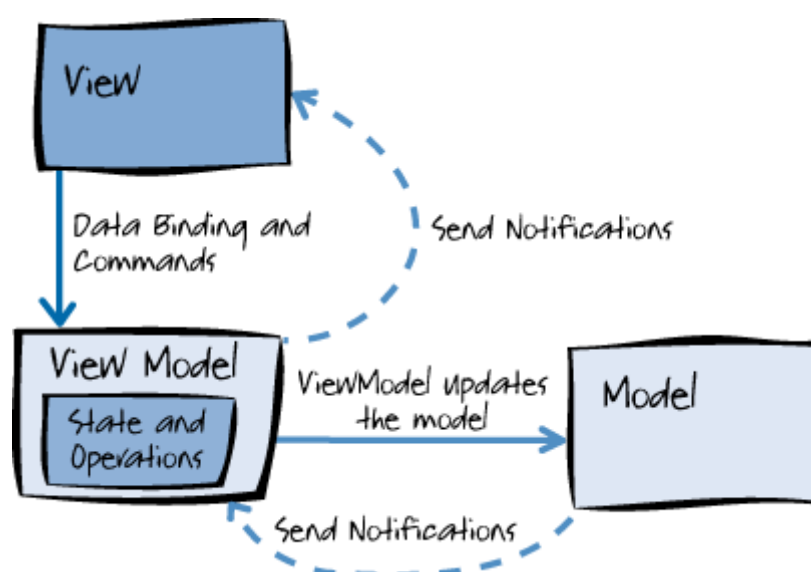
When considering the most appropriate development methods factors including developer knowledge, development time and the suitability of the method were identified as factors contributing to the success of the study. The analysis system to be developed has clearly defined tasks such searches, sorts and classification of data, implementing it using functional programming using the F# programming language was considered as a possible solution. F# is available within the chosen development environment and "...the functional approach is advantageous when using it for one or two variants, the OO approach shows its advantages when using it for three or more variants..." (Simon & Rosch 2015); although F# warranted consideration "some data mining algorithms that involve mutation of state did not fit the functional programming paradigm" (Savev & Bailey n.d.) and the system design required only a singleton class; the amount of time taken to learn F# and link it to a Microsoft Windows based system would cause a significant delay to the investigation. While procedural programming was considered as a possible paradigm, software metrics in "A Measurement Based Comparative Evaluation of Effectiveness of Object-Oriented Versus Conventional Procedural Programming Techniques and Languages" (Ahmad & Talha 2002) indicated that in all but two out of thirteen software metrics an OOP language out-performed a procedural language in execution speed, consequently this and the developers experience in object oriented principles "the method of choice for both facing complex software-development tasks and accomplishing requirements like reusability" (Simon & Rosch 2015) was the selected methodology to enable the core code to be reused in future investigations.

Java, C++ and C# were reviewed as OOP development languages for the application, research into object oriented languages suggested that "C# classes are significantly more likely to be coupled than C++ and Java classes through inter-class method invocations instead of direct data access." (Wu et al. 2015) which supports the loose coupling objective of the development. Java was considered based on portability across



operating systems but C# offered a more familiar user experience to the test subjects aligning with the user-friendly objective of the study.

The Model-View-ViewModel (MVVM) design pattern was chosen to aid implementation; it has a significant learning curve but “code is easier to understand, maintain and troubleshoot” (Tewari 2013) and provides “a clean separation of concerns between the user interface controls and their logic.” (Microsoft 2012) as shown in Figure 22-Model-View-ViewModel Principles (Microsoft 2012).



The components are decoupled from each other, thus enabling:

- Components to be swapped
- Internal implementation to be changed without affecting the others
- Components to be worked on independently
- Isolated unit testing

Figure 22-Model-View-ViewModel Principles (Microsoft 2012)

The model provided an easy system for developing the core functionality of the program without reliance on the user interface, this enabled the analyse class to be built and tested in the early stages of development. The user interface can be developed knowing the model is robust reducing the likelihood of error, for example if output changed when the view model is attached to the model the error could be identified as being caused by the state and operations in the view model.

**Objective A - Is Moodle capable of providing the tracking data in a suitable format to be further processed?**

As part of objective A, it was necessary to identify if the software would have a tangible reason for being developed. A short survey was created to identify: if resources on the VLE were vital to a student's success; what elements of the VLE were being used; what existing processes were performed to evaluate course design and whether designers considered themselves capable of extracting data from the VLE. The Google forms system was selected which allows the "production of a survey with a limited set of question types, immediately pushes survey responses into a Google spreadsheet" (Greenhow et al. 2009); allowing for easy processing of the responses. The questions can be found in section 8.3.1 Moodle usage survey in Appendix C –Surveys. The results of the survey are to be used to decide if the research has a purpose for the course designers and to aid in the selection of the data to be extracted from the Moodle VLE; a full discussion of the results can be found in Section 5.2.1.

To identify if Moodle can provide the tracking data in a suitable format, the Moodle Configurable Reports plug-in shown in Figure 23-Moodle configurable reports plug-in was investigated as a solution to extracting data.

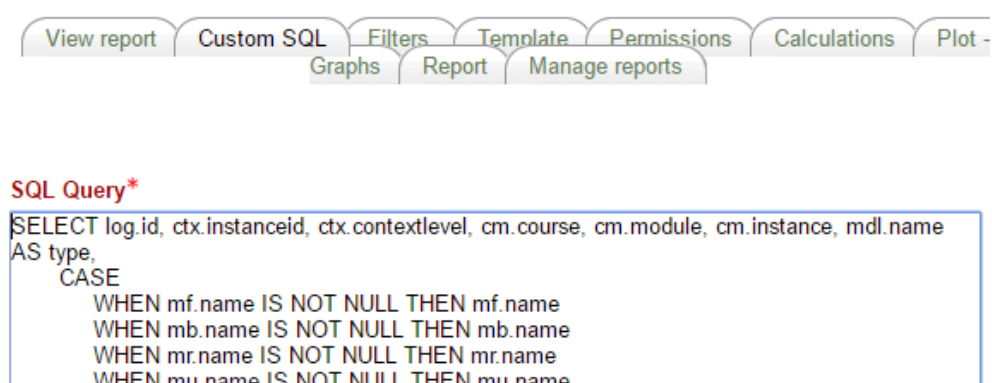


Figure 23-Moodle configurable reports plug-in

The plug-in allows the export of data from the Moodle database in a few formats that can be used with common spreadsheet and database software. The OpenDocument Spreadsheet (ODS) file type was selected, it provides “the broadest support of all formats within the industry” (OASIS 2007). The development initially requires the production of a SQL query within the plug-in to retrieve the usage data for a course from Moodle. Extraction of the data via SQL is challenging due to the complexity of the Moodle database design and the lack of usable official documentation. The construction of the query requires the reverse engineering of the Moodle database using an online, free for educational use diagramming tool lucid chart available for use online at <https://www.lucidchart.com/>. The tool provides a set of SQL commands that dump a database structure into a format that can be imported into lucid chart for further manipulation; Appendix A–Moodle engagement plug-in ERD is an example of the output produced. Analysis of the relevant subset of tables in the Moodle database structure is necessary to identify and understand the occasionally unusual relationships between the tables within the Moodle database as shown in Figure 24-Simplified Moodle tables and relationships used by query, which identified that a link table is not used to relate the instance id in the course module table to the module table.

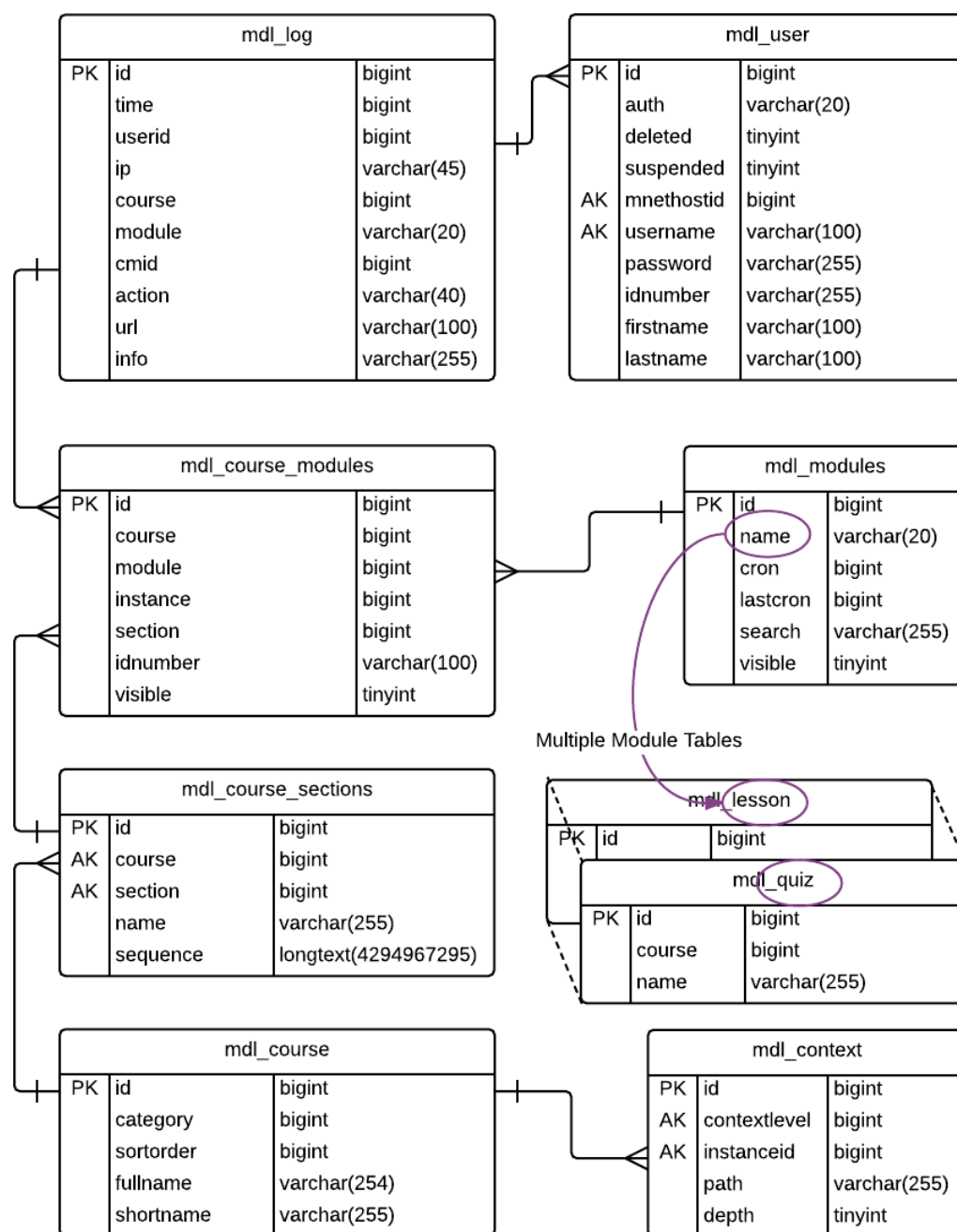


Figure 24-Simplified Moodle tables and relationships used by query

Selecting the correct module table in the Moodle VLE code is achieved through scripting rather than SQL, with the Modules table name attribute corresponding to the name of the Moodle table holding the module data. Table 2-Columns used in the log extraction query shows the full set of columns used in the query with the columns used to resolve the table relationship highlighted.

Moodle Table	Column	Description
Log	Id	Unique log id
Log	Time	Date/time of log entry
Log	Action	Type of user action
Log	url	Moodle URL of action
Log	Userid	Id of user performing action
Context	Instanceid	Id of a context entry for a specific item
Context	Contextlevel	Context in the system
Course Modules	Course	Id of the course linked to the module
Course Modules	Module	Id of the module
Course Modules	Instance	
Modules	name AS type	The name of the module type e.g. quiz
<i>n</i> Module tables	name AS description	Name from table <i>n</i> in query case statement.
User	Firstname	Users first name
User	Lastname	Users last name
User	Username	Users Moodle username
Course	Fullname	Long name of the course
Course Sections	Section	Id of the section
Course Sections	Name	Long name of the section
Course Sections	Sequence	Comma separated list of activity module ids in display sequence order.

Table 2-Columns used in the log extraction query

To enable the query to function requires a SQL Left join to a separate table for each type of activity module to be tracked and an associated case clause to substitute a value into the description column of the results set. The full detail of this method is shown in Appendix B-Log extraction query with sample output. To restrict the results to log entries relating to activity modules that form the interactive elements of the system as shown in Figure 25-Moodle context structure(Moodle Docs 2011), required the use of a specific context value "A context is a "space" in the Moodle, such as courses, activity modules, blocks etc." (Moodle Docs 2015b) an activity module is represented by the value 70.

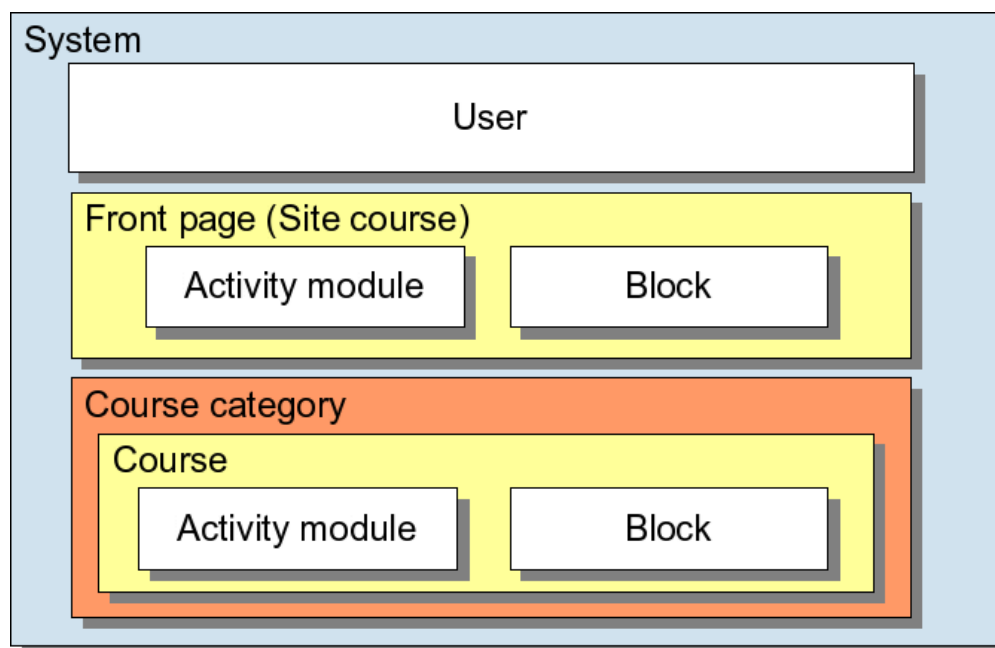


Figure 25-Moodle context structure (Moodle Docs 2011)

Filtering can be applied through the plug-in shown in Figure 26 to allow the selection of a date range for the log file, achieved through the plug-ins' replaceable parameters in the query.

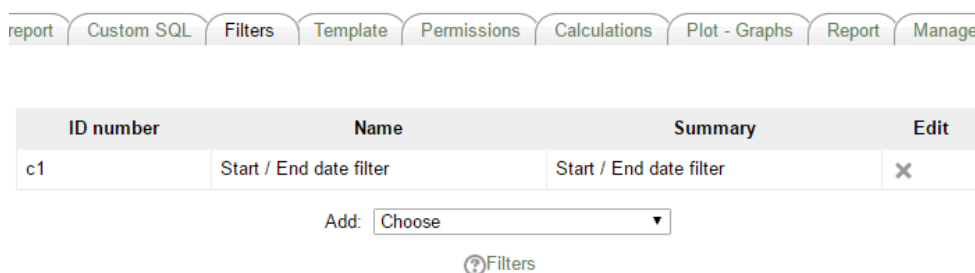


Figure 26-Adding a date range filter to a Moodle configurable report.

This provides a filter block in the report results section, linking to the query where clause as shown below, where time is the time column from the Moodle log table.

WHERE %%FILTER\_STARTTIME:**time**:>%% %%FILTER\_ENDTIME:**time**:<%%

The results of the query are downloaded to a local ODS file suitable for processing in objective B.

**Objective B - Can effective processing be performed on the data to provide meaningful information?**

As a result of development undertaken in objective A, an ODS data file has been created, the development of a C# Model class (Analyse) for the manipulation of the data is to be developed using MVVM techniques. An open source utility class ODSReadWrite (GemBox Software 2011) along with an associated dynamic link library (DLL) DotNetZip (Jaans 2012) provides the capability to read and write ODS files. The class is to be developed from a simple use case based on the following requirements, for the system:

1. File access.
  - a. Read Moodle ODS.
  - b. Read student list and grades(ODS).
  - c. Write student list and grades (ODS).
2. Data processing.
  - a. Find and collate module access (Module List).
    - i. Sort modules by Moodle course sequence.
    - ii. Sort modules by number of unique accesses.
    - iii. Sort modules by number of accesses.
    - iv. Find the highest module access count.
  - b. Find and collate section information (Section List).
  - c. Find and collate student information (Student List).
    - i. Select all students.
    - ii. Deselect all students.
    - iii. Invert student selection.
    - iv. Select students by grade criteria.
    - v. Count active students.
  - d. Clear data.
3. MVVM behaviours
  - a. Notify property change.
  - b. Property change event handler.

Prior to the development of the user interface a loosely coupled reusable 'Analyse' class is to be developed and tested using unit tests, establishing operations were being performed correctly meeting the functional requirements of the system.

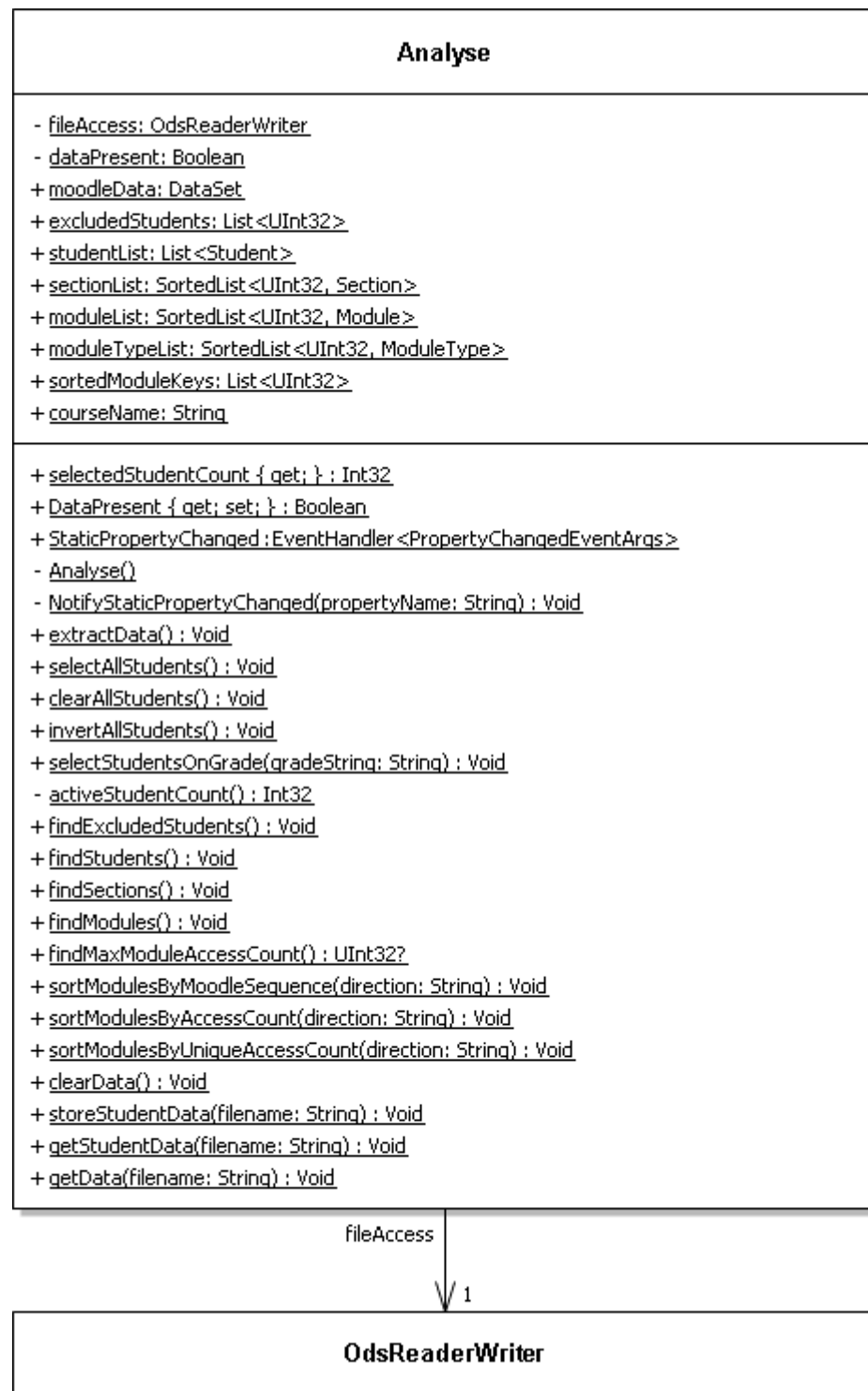


Figure 27-The ODS Analyse class

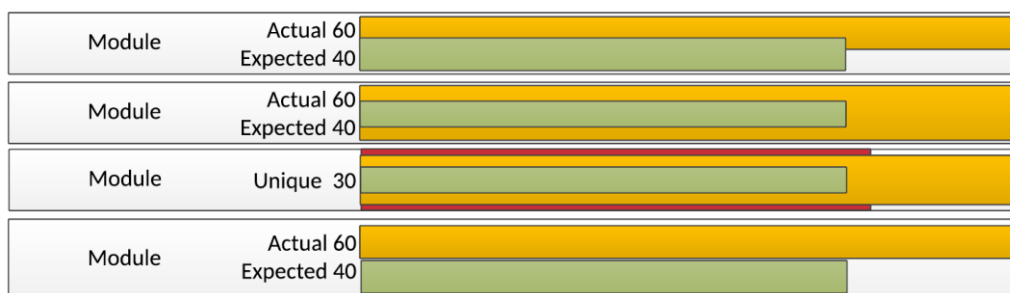


**Objective C - Can the result of the processing be displayed in an easy to interpret format for a designer to use?**

Paper prototyping is used in the development of the software development methodology to ensure the software created to raise acceptance of data mining in course evaluation was sufficiently useable and understandable in the context of a video demonstration. User feedback is to be acted upon to ensure that software was not rejected by the users as ineffective due to usability problems.

The focus is initially on the best model for presenting the information, a bar graph was selected as the most appropriate tool for presenting the usage data, given that “bar charts are useful for displaying data that are classified into nominal or ordinal categories” (Student Learning Development 2009) and that the data would need to be sorted into more than one sequence (Moodle course order, total accesses and unique student count).

Four separate designs for the bars were presented to existing Moodle users to establish the most understandable representation.



Feedback indicated the bar with a red minimum guide as the most understandable version and was integrated into the methodology for use in the early stages of the development.

The initial prototype was developed in lucid chart using its user interface design template Figure 28-Initial user interface prototype MooLog.



Figure 28-Initial user interface prototype MooLog

The first design was rejected as being cluttered with too few students visible; consequently, productivity would be hampered when selecting students to analyse and it would be hard to interpret the bar chart if there were too few bars visible. In response to the user feedback the system was segregated into a two-window based system, a main window for user selection and file management and a reporting window allowing the selection control and the display of the data elements a greater amount of screen space as shown in Figure 29-Final user interface MooLog.

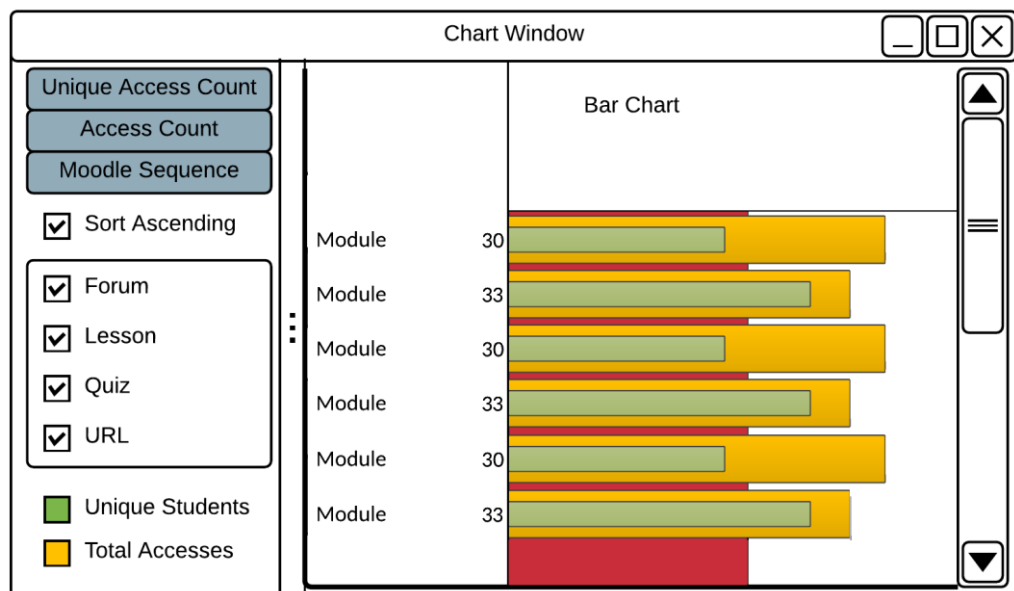
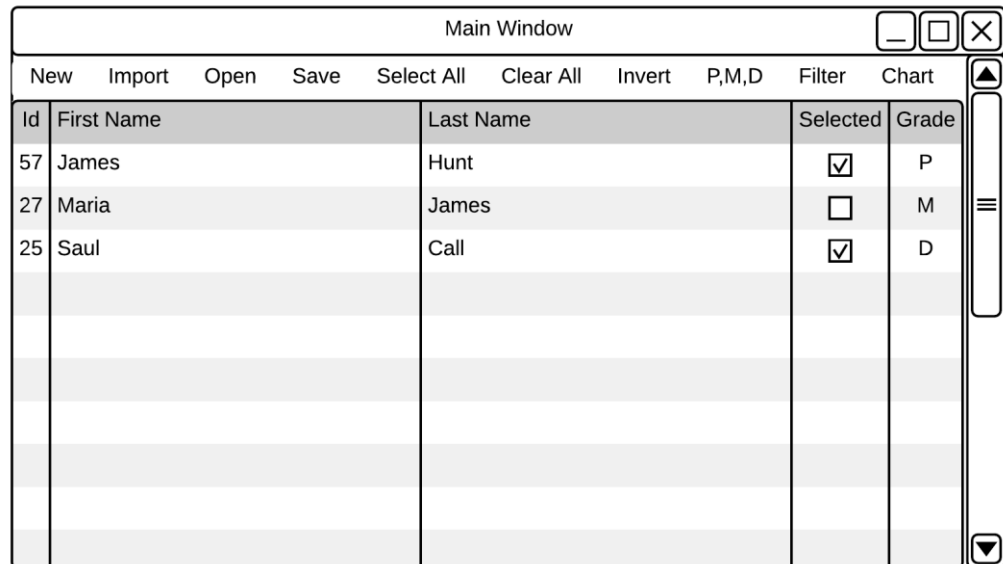


Figure 29-Final user interface MooLog

With the user requirements satisfied as part of the methodology the use of Microsoft Visual Studio 2015 allows the easy transition from prototype to an application user interface through its Blend XAML interface design tool providing the visual element of the application using the Windows Presentation Foundation (WPF) framework for interaction.

The main student selection window shown in Figure 30 was created first within Blend to enable the selection of ODS files and filtering of students to be analysed. This stage required the production of WPF code to act as the data binding between the user interface and the analyse class.

Id	First Name	Last Name	Selected	Grade
22	A	Student	<input checked="" type="checkbox"/>	D
16	A	Student	<input checked="" type="checkbox"/>	P
20	A	Student	<input checked="" type="checkbox"/>	P
37	A	Student	<input checked="" type="checkbox"/>	P
28	A	Student	<input checked="" type="checkbox"/>	P
24	A	Student	<input checked="" type="checkbox"/>	D
23	A	Student	<input checked="" type="checkbox"/>	D
7	A	Student	<input checked="" type="checkbox"/>	P
18	A	Student	<input checked="" type="checkbox"/>	M
9	A	Student	<input checked="" type="checkbox"/>	D
30	A	Student	<input checked="" type="checkbox"/>	M
12	A	Student	<input checked="" type="checkbox"/>	P
21	A	Student	<input checked="" type="checkbox"/>	M
34	A	Student	<input checked="" type="checkbox"/>	D
10	A	Student	<input checked="" type="checkbox"/>	M
36	A	Student	<input checked="" type="checkbox"/>	P
27	A	Student	<input checked="" type="checkbox"/>	P
25	A	Student	<input checked="" type="checkbox"/>	P
26	A	Student	<input checked="" type="checkbox"/>	R
11	A	Student	<input checked="" type="checkbox"/>	P
15	A	Student	<input checked="" type="checkbox"/>	P
29	A	Student	<input checked="" type="checkbox"/>	M

Figure 30-Implemented student selection window

WPF separates the appearance of a user interface from its behaviour as shown in Figure 31, the link is formed in the code behind the form using a *CollectionViewSource* object (Microsoft 2015) allowing the list of students in the *Analyse* class to be linked to a data bound grid on the main form that displays the students interactively.

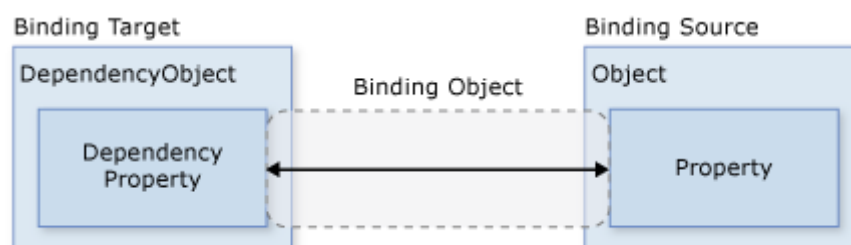


Figure 31-WPF Databinding process(Microsoft 2010)

The use of this feature reduced the complexity of the code relating to the manipulation of student data significantly.

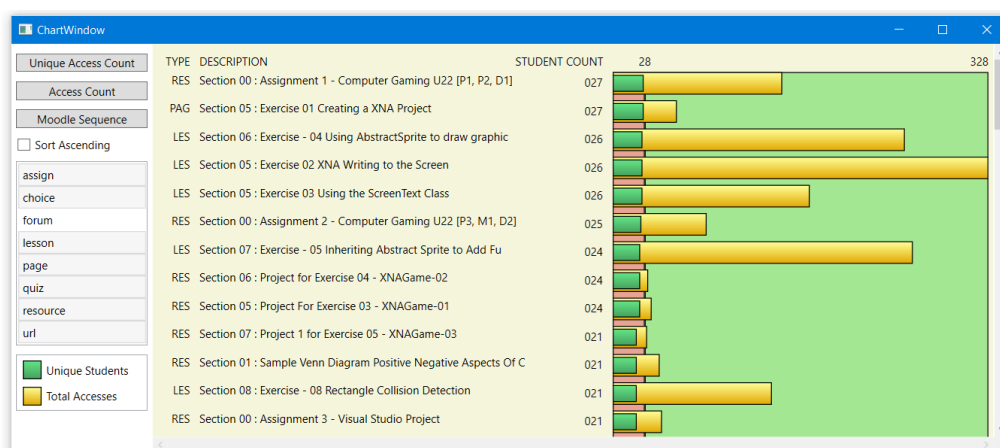


Figure 32-Implemented chart window

**Objective D - Is the information provided by the software useful to the designer?**

Fundamentally the end goal of the research was to establish if Moodle data could be useful to a course designer. Objectives A to C established the mechanisms to perform this through the development of an application that could be used by a Microsoft Windows user with experience of using software such as Microsoft Excel or Word.

A research methodology needed to be applied to ensure that the final application delivered the promise of being useful to a designer. Investigation into research methods suggested that the use of an online survey with video demonstration “can be easily used when we know the basic population and have online access to its members” (Szolnoki & Hoffmann 2013) and “the speed of data collection” (Blair et al. 2013) would facilitate effective collection of quantitative data. The questionnaire would establish the level of success of the application and its potential for adoption by designers. Initially an investigation was made into the users of Moodle in the college to identify the teaching staff that actively engaged with the VLE to ensure the survey was being sent to staff who were regularly modifying courses on the system. The list of subjects was derived by analysing the Moodle access logs, filtering staff that logged in and modified material Moodle at least once over a seven-day period, this process was repeated for four weeks resulting in 58 candidates who consistently engaged with the system. A sample size of 51 was identified as being a representative sample of the 58 frequent users of

Moodle teaching within the college, giving 5% margin of error with a confidence level of 95% (CheckMarket 2015).

The online survey link is distributed to each candidate via email; to ensure a sufficient response rate tutors are contacted the week after the initial email to encourage completion.

The survey design was split into two key areas.

1. Questions to establish existing working practices of the users to identify if they aligned with the software design, prior to evaluating the features of the software.
2. Questions to identify if the software features were of use in the context of evaluating and supporting redevelopment of courses.

The use of interviews was also considered as the primary research method, but was rejected because the study required a representative sample of the users of the colleges' Moodle system to identify if acceptance of the system has been raised rather than 'Why it is not currently used?' this data is quantitative data "offering 'hard', 'factual' data" (Barnham 2015). Further studies may want to explore the 'Why?' change rather than the 'What?' has changed.

### **3.2.3 Evaluation**

The VLE analysis software created for this study is intended to act as an enabler for course designers/tutors; the evaluation intends to focus on the main objective of identifying if providing information in an uncomplicated format through a convenient software tool, would it encourage the staff to utilise such tools in the future as part of their normal activity? The outcome of the evaluation will be an identification of usefulness of the information provided by the features of the system and an evaluation of feedback from course designers regarding their future acceptance of data mining as viable resource evaluation system to improve the quality of courses they provide to learners and a set of recommendations for further research in this area. A successful outcome of this would be to

identify that the software did result in an increased commitment to the use of data mining by the designers compared to their existing utilisation. It is hoped that the consequence of providing a better analysis tool would be that designers would empirically evaluate their course resources and raise the resource quality leading to better engagement by students with the VLE based activities.

### **3.3 ETHICS**

The study is being performed within Lincoln College using students within the Computer Science section, all automated activities will be based on existing data collected by the VLE which is freely available for teaching staff to access. Students will be required to connect to an external version of the VLE but will not be required to enter personal contact information other than their college id, name and social networking public user name. This will not be linked to any personal information within the college systems. Access to the VLE will use <https://> encrypted communications and access to the data in the system is restricted to the student and the tutor responsible for the student.

- Permission for the use of student logs has been granted by the college.
- Any student or staff member involved in the surveys and log analysis will be asked to give consent prior to the use of the system.
- Any student wishing to opt out of the trial can do so by removing their contact details from the VLE.
- All social networking accounts attributed to students will be verified before messages are sent to the student for the first time.
- The data collected will be altered to ensure anonymity of the students.

## 4 DATA ANALYSIS

---

### 4.1 STUDENT ENGAGEMENT

Student engagement was measured by their timeliness of access to the resources on the VLE; a student was determined to be appropriately engaged with the material if they accessed it either before the expected access date or within a window of two weeks after the date. If a student failed to access the material after that date they were considered not to be appropriately engaged with the materials.

Prior to interpreting the quantitative data from the log files statistical tests were undertaken to ensure the results were statistically significant. IBM SPSS software was utilised to extract relevant results and an additional calculation for identifying the effect size.

Performing the statistical tests required some pre-processing of the captured data using the criteria identified below.

- Positive accesses  $\leq$  2 weeks of expected access date.
- Negative accesses  $>$  2 weeks of expected access date or no access made.

The groups were split into the categories

- Tracked – the experimental group who were prompted to engage.
- Untracked – the control group.



#### 4.1.1 Independent Samples t-Test

	Tracked	N	Mean (X)	Std. Deviation ( $\sigma$ )	S.E. Mean
On Time Access	Yes	900	.54	.50	.02
	No	840	.33	.47	.02

		Levine's Test for Equality of Variances		t-test for Equality of Means						
On Time Access		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
	Equal variances assumed	94.60	.000	9.23	1738.00	.000	.21	.02	.17	.26
	Equal variances not assumed			<b>9.24</b>	<b>1737.77</b>	<b>.000</b>	<b>.21</b>	<b>.02</b>	<b>.17</b>	<b>.26</b>

An independent-samples t-test was conducted to compare the timeliness of VLE accesses in tracked (electronically prompted to engage with missed activities) and untracked conditions. There was a significant difference in the scores for prompted ( $M = 0.54$ ,  $SD = 0.50$ ) and unprompted ( $M = 0.33$ ,  $SD = 0.47$ ) conditions;  $t(1738) = 9.24$ ,  $p = 0.0002$ .

These results suggest that electronic tracking and prompting of students who miss activities does have an effect on the timeliness of their engagement with the VLE activities. Specifically, the results suggest that when students receive a Twitter

direct message to engage with missed material, they complete missed activities closer to required completion date.

A small to medium (0.43) effect size (Cohen 1992) was identified when the Cohens d calculation was applied to the results.

$$\text{Effect Size Cohens } d = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{2}}} = 0.43$$

The experiments hypothesis indicated that students who were tracked would engage more than non-tracked students, given that T value is greater than critical value 1.96 obtained to from table B2 (Salkind 2014, p.372) the null hypothesis that both groups would be equally engaged cannot be accepted.

#### 4.1.2 Comparison between accesses

From the results of the t-test and Cohens d it was identified that the hypothesis for students being prompted accessing the materials earlier was most likely and that the effect size was close to being classified as medium.

From Figure 33 - Graph of student access profiles it can be seen that the positive accesses for the experimental group was 22% higher and they completed 19% more activities than the control group. The experimental group also performed 3% better than the control group on late accesses.

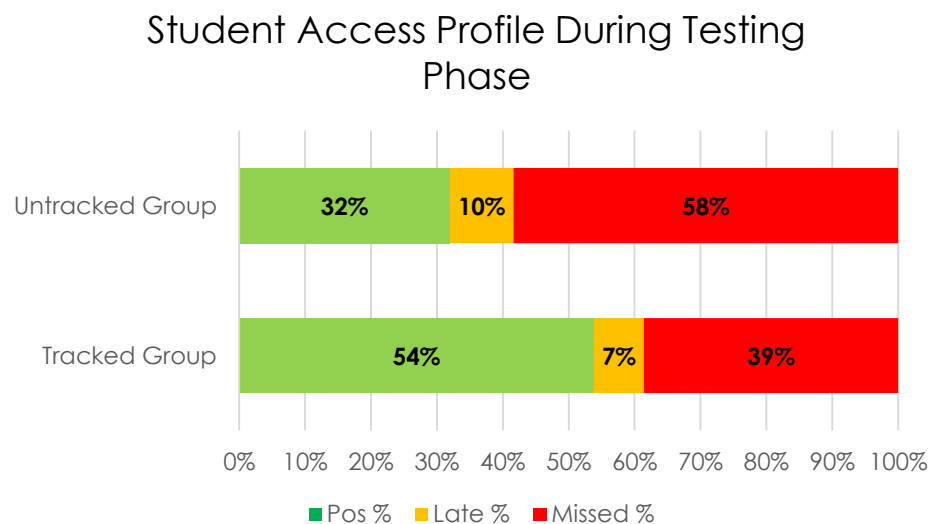


Figure 33 - Graph of student access profiles

The high levels of late and missed activities by the untracked group provides a strong argument for the need for tracking of student engagement given that the tracked activities were flagged as mandatory elements of the courses.

#### 4.1.3 Positive access trend comparison

Over the duration of the experiment the prompting was applied multiple times to the experimental group to identify if the process would show a positive or negative cumulative effect on them in comparison with the control group. Miltenberger in the area of prompting and transfer of stimulus control indicated that through multiple prompts the engagement should increase (2011). From Figure 34 - Engagement trends during study,

it has been observed that the experimental group showed a positive trend in the mean timeliness of access of 16% from the start of the experiment indicating that repeated prompting does have a positive cumulative effect over time.

In comparison, the control groups' improvement over time has not increased and is 34% lower than the experimental group by the end of the study.

### Tracked vs Untracked student positive accesses over the duration of the study

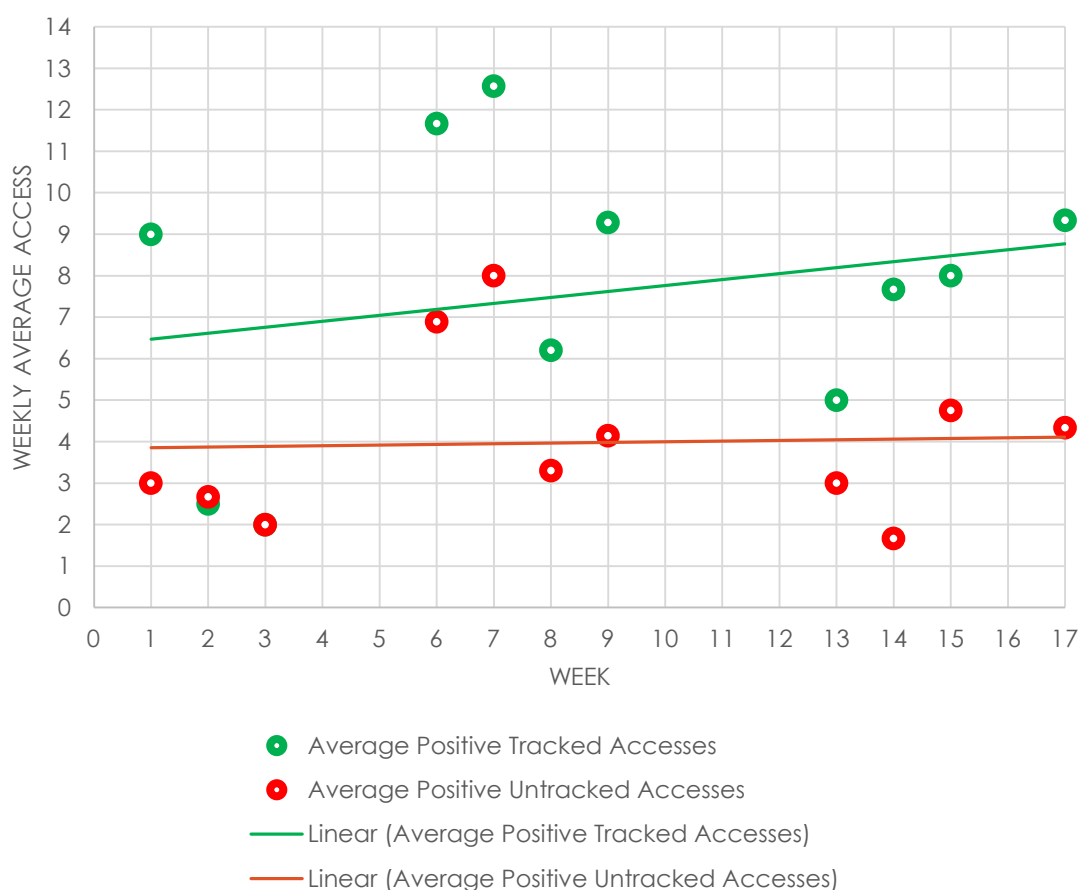


Figure 34 - Engagement trends during study

When evaluating the results, it was noted that the untracked group showed an initial lower level of performance and it remained so throughout the study compared to the tracked group; this does not however, relate to how the behaviour of students was altered by prompting as the study compares the improvement from each groups

start point. In Figure 34 the trend of the prompted participants showed continuing improvement from the participants starting point compared to the static results of unprompted students giving a good indication that there was an increase in engagement in the experimental group because of the messages being sent to them. The data points in weeks 6 and 7 on the graph indicate a higher level of completions by both groups, this can be explained by the delivery within those weeks being lighter around the half-term break within the 15 weeks of study, allowing a greater opportunity both in class and during the week break for some tasks to be completed.

## 4.2 VLE USAGE ANALYSIS SOFTWARE

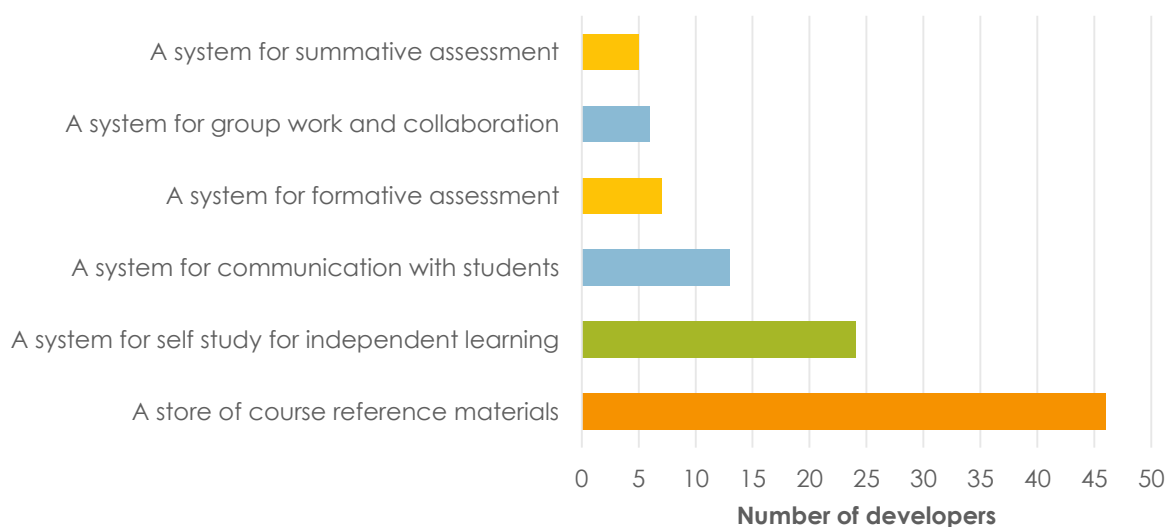
Data for the course analysis tool was collected using Google forms based surveys, the surveys collected responses from 50 Lincoln College employees who were actively using the colleges VLE system. The questions in the surveys related to the current VLE usage and the utilisation of the course analysis tool aid in the analysis of VLE course materials and their utilisation by students.

### 4.2.1 Pre-development

The participants were surveyed to establish if there was a purpose to the development of the course analysis tool and to identify existing VLE usage.

#### ***How developers use Moodle features***

To ascertain what elements of Moodle were most likely to need tracking developers were asked to identify their most frequently used elements.

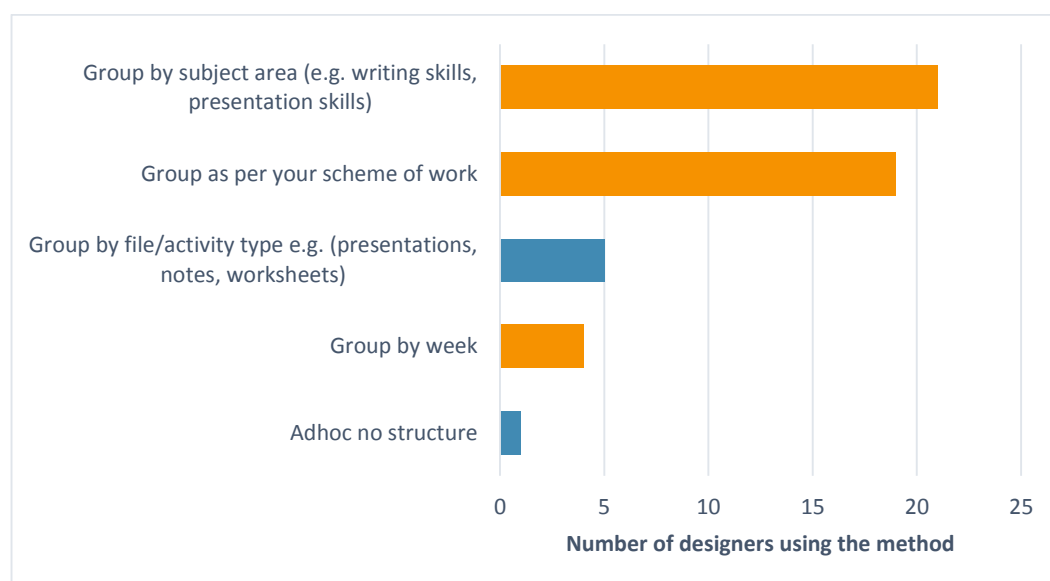


Out of the 50 course developers used Moodle:

48%	as a system for student self-study
92%	as a store of reference material
25%	for assessment
36%	for communication and collaboration

### **How developers organise resources**

Developers were asked how they organised resources on Moodle to help identify if there was any specific structure in the organisation of materials.



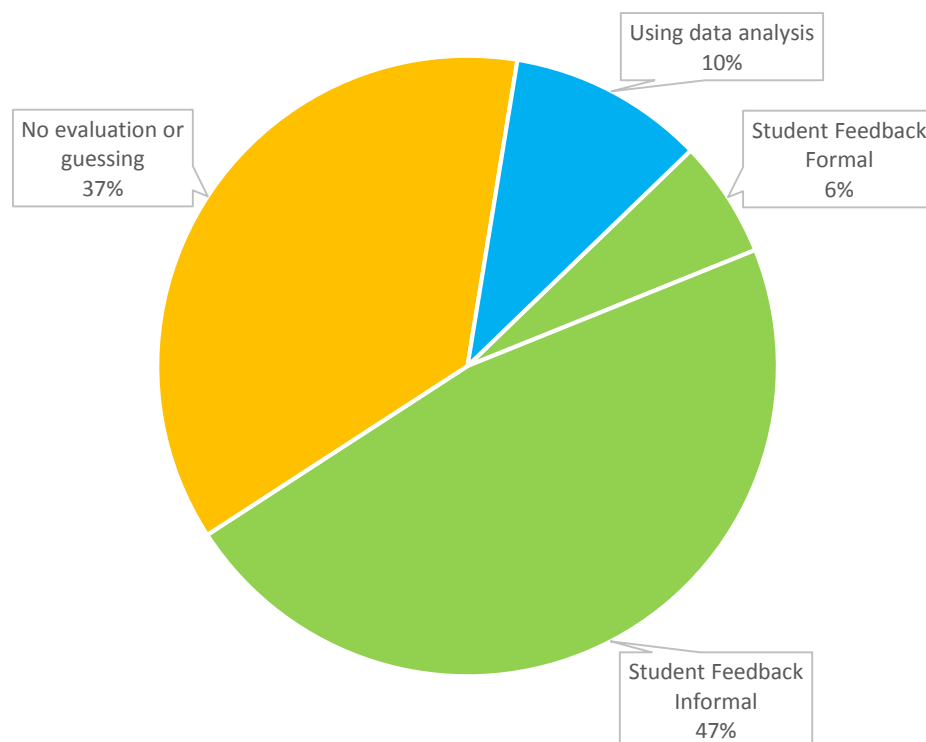
Of the 50 developers:

88%	structured their materials in relation to topics or planned delivery sequence
12%	did not relate the materials to a student's study sequence.

This indicated that most developers considered how the materials were to be accessed by the users, identifying there was some level of planning and structure to VLE courses within the college.

### ***How designers evaluate their course design***

To establish what systems designers already applied to when evaluating a course, they were asked to identify their existing mechanisms.



Of the 50 developers questioned about how they evaluated:

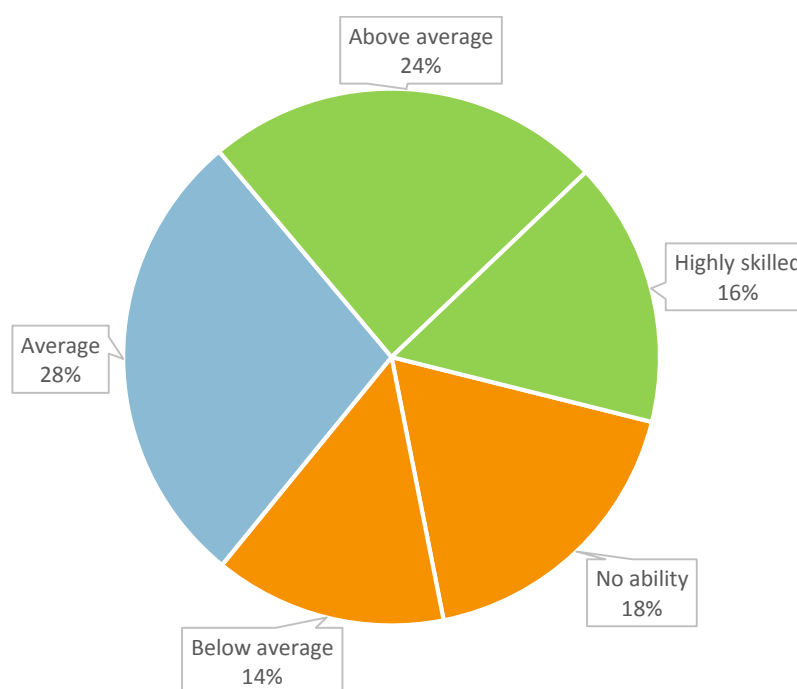
37%	did not evaluate or guessed
53%	used student feedback
10%	used data

The results suggest that although designers previously indicated that they structured their VLE courses, a significant proportion of the designers did not evaluate the courses they created or used informal feedback to influence improvements. Very few designers stated they used data analysis, possibly indicating that this method is either unfamiliar, perceived as difficult or they lack the skills to use the method.



### **Developers ability to gather data about student activity on Moodle**

To establish the course developer's basic skill level in extracting data from the standard Moodle user interface, they were questioned about how able they were at retrieving information about a student's activity on the system.



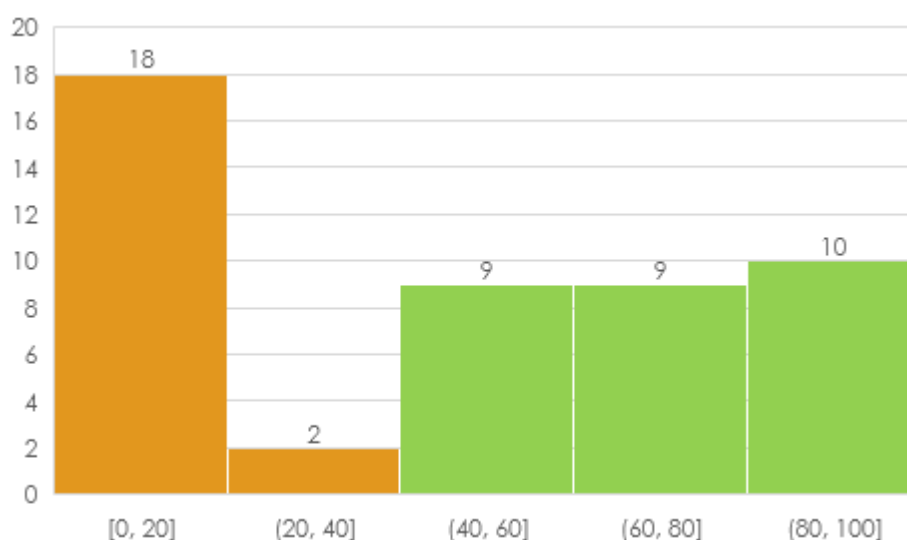
Of the 50 developers questioned:

32%	were below average or without ability at retrieving data
28%	had an average level of ability to retrieve data
40%	were above average or highly skilled in retrieving data

The previous chart indicated that only 10% of designers used data as a method of evaluation and it was proposed that factors that might explain this would include a lack of ability to retrieve the data. The results presented above indicate that for 60% of the designers rated themselves as having from no ability to an average ability to retrieve data. Potentially this may be linked to the low usage of data analysis for evaluation and supports the argument for the use of simple tools to aid tutors in retrieving and analysing data.

**Percentage of materials and activities are vital to complete a course?**

To establish the level of importance of the online resources provided to the students by the course designers, the developers were asked what proportion of a Moodle courses content was vital to the students being successful on their course.



Of the 50 developers questioned:

40%	considered below 40% of their resources as vital
56%	considered at least 40% of their resources as vital
4%	did not answer the question

The results from the question indicated that online resources are in many cases vital to a student completing successfully; at least half of the course designers provided a large proportion of essential online resources within their courses. With a high proportion of resources being vital, there is a need to evaluate them to identify if they are effective and being used by students, this supports the studies aim to encourage evaluation through data analysis.

#### 4.2.2 Post-Development The usefulness of the information provided by the software

The participants were surveyed to identify which of the functional parts of the software would aid them in evaluating the usefulness of elements in a course based on student utilisation.

##### *Knowing how many of your student group accessed each resource*

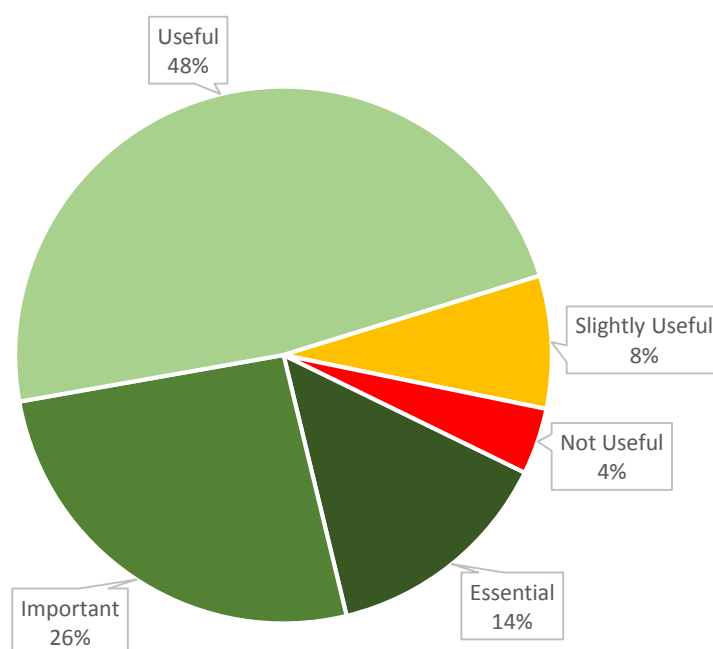


Figure 35-Chart of responses on the importance of knowing how many of a group accessed each resource.

Of the 50 respondents 100% responded with valid responses; 88% of the respondents responded positively, 12% negatively to the usefulness of knowing the total number of individual students that accessed a resource.

The high positive response level gives some confidence in the usefulness of the tool in general with the usage data being provided to course designers being seen as an effective metric.

**Knowing how many total accesses there were to a Moodle resource (popularity).**

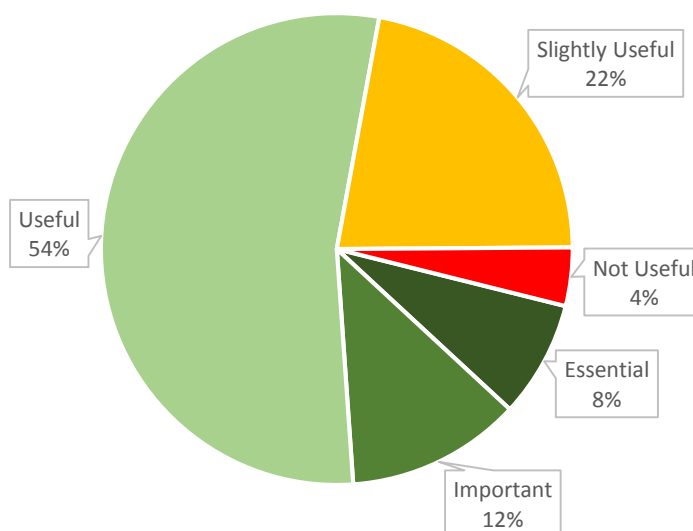


Figure 36-Chart of responses on the importance of knowing how many total accesses there were to a Moodle resource (popularity).

Of the 50 respondents 100% responded with valid responses; 74% of the respondents responded positively, 26% negatively to usefulness of knowing the total number of individual times a resource was accessed.

The popularity of a resource was identified as another useful metric based around usage data, this has a bearing on the level of acceptance of data mining for course evaluation by the designers; showing that data when provided to designers has a very low rejection rate.

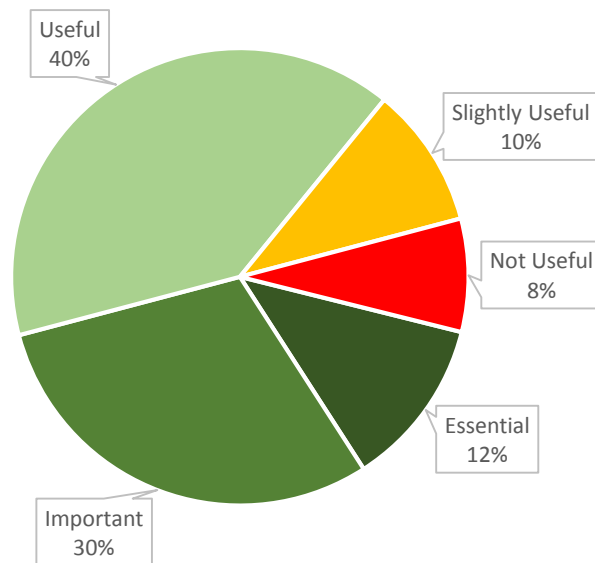
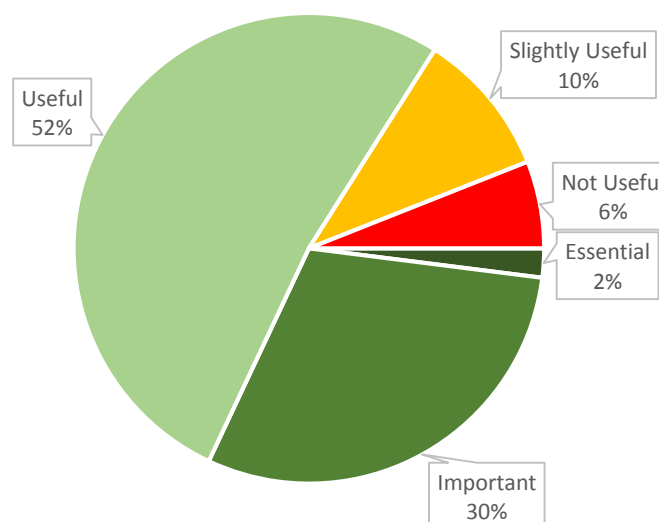
**Being able to identify resources that had low or zero usage.**

Figure 37-Chart of responses on the importance of being able to identify resources that had low or zero usage.

Of the 50 respondents 100% responded with valid responses; 82% of the respondents responded positively, 18% negatively to usefulness of knowing which resources had a low or zero access count. A high level of positive response to the ability to be able to identify low usage resources was expected in this question given that to evaluate the success of a resource a key indicator of a poor resource would be that it is not being used. It was expected that the positive responses would be greater than the responses from the previous two questions, this was not the case, raising questions about the negative respondents; there is some indication that there is a core of respondents that have rejected the application facilities in general.

**Being able to identify resources that were used based on a student's level of achievement,**



*Figure 38-Chart of responses on the importance of being able to identify resources that were used based on a student's level of achievement.*

Of the 50 respondents 100% responded with valid responses; 84% of the respondents responded positively, 16% negatively to usefulness of linking the students level of achievement with the resources they accessed. This question focused on providing linked information (grades to usage) compared to the previous questions that focused on single data items. A greater number of respondents saw this information as important compared to the basic utilisation information from the first two questions. This is an encouraging sign given that this type of information would require a high level of ability and would take a significant amount of time to produce without software and gives some confidence that the survey is identifying positive feedback from the designers.

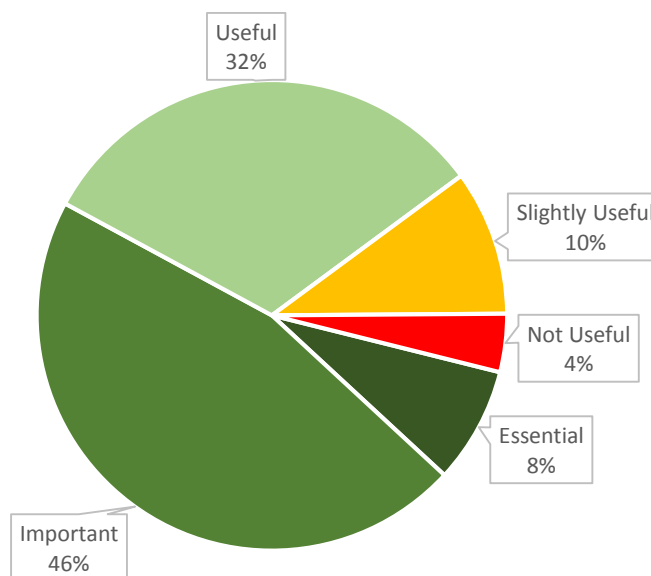
**Being able to find an individual students usage.**

Figure 39-Chart of responses on the importance of being able to find an individual student's usage.

Of the 50 respondents 100% responded with valid responses; 86% of the respondents responded positively, 14% negatively to usefulness of knowing individual student's access to resources. This question focused on identifying the level of importance of being able to mine data about individual students, this question was deliberately placed within the survey to encourage the participants to consider data mining as more than just a course evaluation tool. The increase in responses in the important to essential range may indicate that many of the designers are still more concerned about tracking what students are doing than evaluating the activities they provide to them.

### 4.2.3 Would the tool be used if provided in normal activity?

The participants were questioned about the usefulness of the software when applied their working processes. The specific question asked was:

If this software was made available to you to help you decide what resources and activities worked well with your students, would you use it?

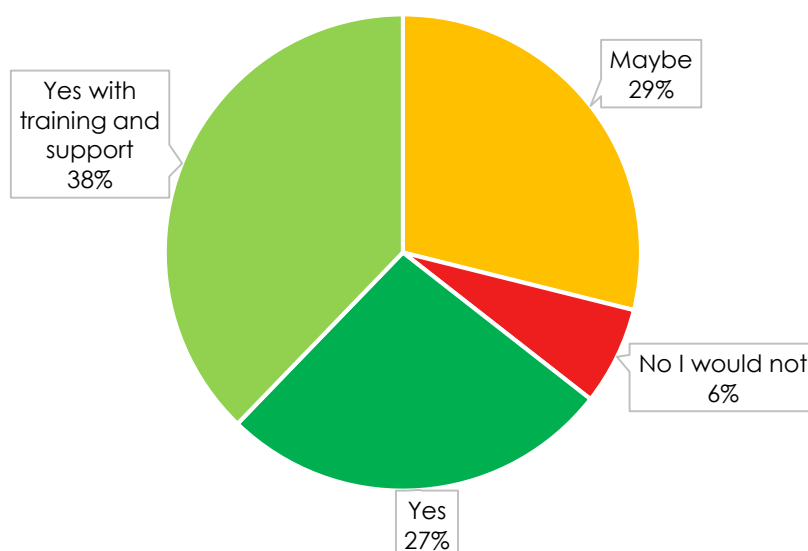


Figure 40-Chart of responses identifying whether the analysis software would be used by the participants.

Of the 50 potential respondents 50 responded with 5 providing null responses. Of the 45 valid responses 65% responded positively, 29% were uncertain and 6% would not use the software. The average positive responses of 83% from the previous questions relating to the usefulness of the data combined with the 65% overall positive response to this question shows there is a clear shift in acceptance of the data mining tool and the data it provides. There does seem to be a core element of responses in the range of 4% - 8% in each question where data mining is rejected; this is low rejection rate and does suggest that overall there would be a good uptake of data mining in future.



#### 4.2.4 Comparison of levels of acceptance between designers who previously did/did not evaluate their courses.

Prior to the demonstration of the software designers were asked if they currently evaluated their courses, this resulted in a 37%(No)/63%(Yes) split within the group. Subsequent to the demonstration all participants were asked if they would adopt the software as part of their normal activities, the chart below identifies the level of adoption of the software by the two groups.

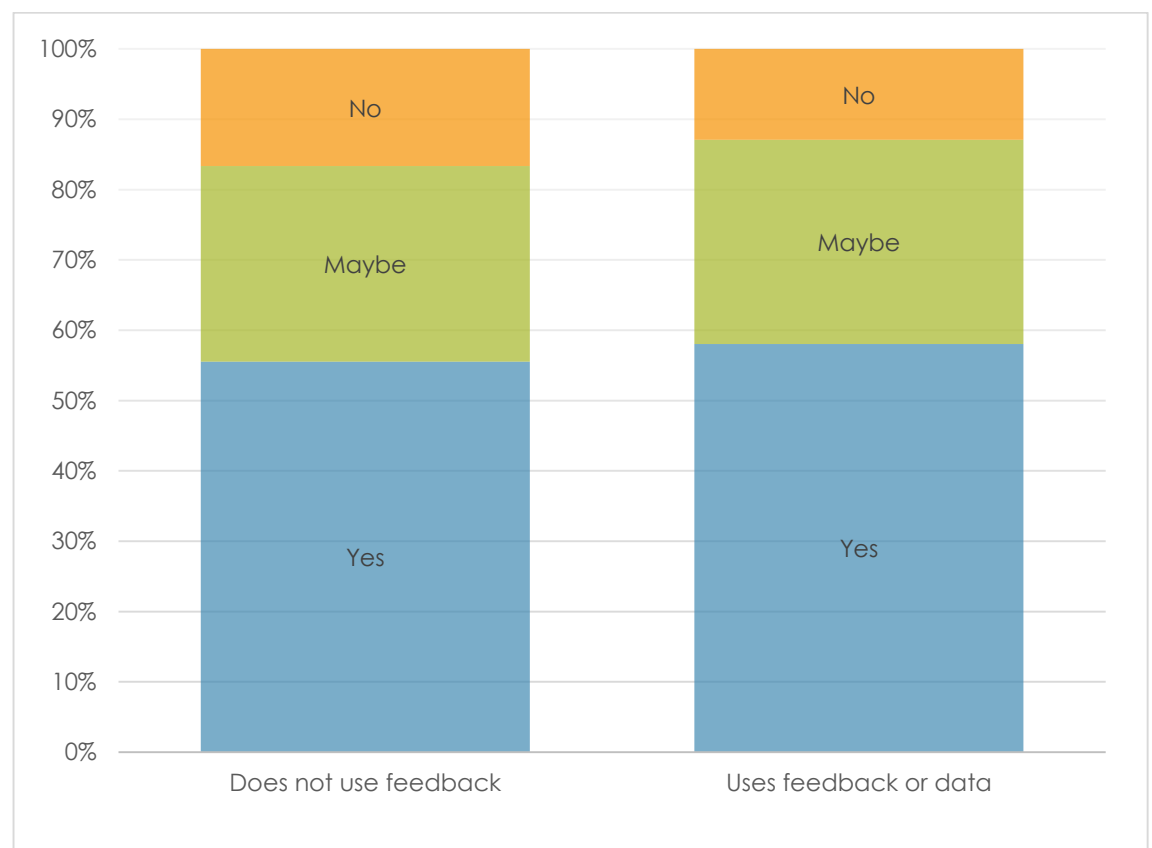


Figure 41-Chart of software adoption by designers who previously did/did not evaluate their courses

The level of acceptance in the two groups showed comparable levels of acceptance of the software as part of their normal design and evaluation activities.

## 5 DISCUSSION

---

This chapter discusses the results of the study described in Chapter 3 Research Methods, the research concentrates on two separate groups of stakeholders: the students who utilise Moodle as a study system and academic staff who deploy resources on the Lincoln College Moodle VLE.

The research consisted of two separate investigations to identify if it is possible to create software to access data held within the Moodle VLE to:

1. improve student's engagement with the system;
2. aid academic staff in understanding the utilisation of the resources placed on Moodle to facilitate course evaluation and design.

The discussion is split in to two separate sections: 5.1 Student engagement system and 5.2 VLE usage analysis software to assist with clarity.

Each part of the research methodology required the development of software prior to gathering data, consequently the software development process is also discussed in these sections.

### 5.1 STUDENT ENGAGEMENT SYSTEM

Research into learner engagement with Lincoln College's Moodle VLE system was undertaken to gather empirical data relating to the level and timeliness of students accesses to course materials and to establish the impact of using a self-devised software automation tool to encourage less engaged users of the system to access resources in a timely fashion.

The literature review revealed that use of stimulus of control could be a method of changing student behaviour as identified by both Miltenberger (2011) and Bloh (2008), and that Mohamad & Tasir (2013) suggested social networking could be a possible tool to use to provide the initial stimulus.

Subjects selected for the study were on the second year of a BTEC Extended Diploma in Information Technology course at Lincoln College; two units from the course were involved in the study, developing computer games and website development. For the purpose of the study the students were split into two groups; a control group and an experimental group.

The hypothesis was that the experimental group (students receiving automated prompting) would access missed VLE course materials sooner than the control group.

The null hypothesis was that the experimental group would not access the learning activities any earlier than the control group after being prompted.

The development of the Moodle engagement plug-in (software tool) formed a part of the methodology for the research, controlling both the identification of students who were not engaging and instigating the Twitter and support email communications with the experimental group.

The source of the empirical data was the exported log files from the Moodle VLE outlined in Figure 19-Entity relationship diagram exported log data in Section 3.1.7 Student engagement system evaluation. The data allowed the identification of individual student resources access times against the expected times of access.

The study was conducted over 15 weeks with 57 tracked deadlines being monitored by the Moodle engagement plugin. The results for each participant were categorised in to three outcomes:

- Positive – access occurred within two weeks of a deadline.
- Late – access occurred after two weeks of a deadline.
- Missed – no access made within the scope of the course duration.

The statistical significance of the experimental data compared with the control group was established through an independent-samples t-test

detailed in Section 4.1.1 Independent Samples t-Test. The results of the t-test suggested electronic tracking and prompting of students who miss activities does have an effect on the timeliness of their engagement with the VLE activities. The size of the effect based on a Cohens d calculation was identified as being a small to medium effect size.

Specifically, the results suggest that when students receive a Twitter direct message to engage with missed material, they complete missed activities closer to required completion date.

The comparison between the accesses of the control group against the experimental group revealed a 22% greater positive accesses to the online course resources by the experimental group and that overall the control group accessed 19% fewer resources than the experimental group.

Further analysis of the timeliness of access by the control and experimental groups over the duration of the study in Section 4.1.3 Positive access trend comparison found that the experimental group demonstrated an increasing trend in prompt accesses at a greater level than the control group. The findings suggest that repeated prompting leads to improvement in timely engagement with resources on the VLE which is in agreement with Miltenbergers (2011) work on the principles of stimulus control.

Overall the data analysis of the results of the study showed a positive increase in performance by the experimental group. The results of the study suggest that the use of a software solution to create an electronic message as a stimulus to produce a change in operant behaviour was successful within the parameters of the experiment effectively rejecting the null hypothesis was that the experimental group would not access the learning activities any earlier than the control group after being prompted.

When reflecting on the success of the research there is a need to review the limitations of the study including the use of a small sample of

computing students within one institution which may not reflect the range of learners with the larger academic community. The duration of 15 weeks for the study may not have allowed time to establish if the performance gains made by the experimental group could be further improved by the continued use of the system or would the learners become disillusioned with system and cease to respond to prompts.

The use of extracted quantitative data from the Moodle database gave clear empirical evidence to establish the student's activity and to identify changes in behaviour and was an effective research methodology in this context simplifying data collection and processing.

The extraction of existing data from Moodle database through the use of a plug-in was effective within the scope of the experiment and validates the existing research by Mazza et al. (2012) and Zhang & Almeroth (2010) that focused on the extraction and reporting from the system.

It is important to note that the plug-in development process was hampered somewhat by the limited documentation relating to the structure of the data held within the system and omissions in the open source Moodle documentation. Despite the development issues and that Moodle is an open source project, the plug-in system proved to be completely stable and performed reliably for full duration of the research suggesting it should be considered a viable system for conducting research on student's utilisation of the Moodle VLE.

Early in the research the literature review identified that while Mohamad & Tasir (2013) proposed that the use of social networking was a possible communications mechanism, out of all the social networking providers only Twitter provided an API to allow for direct messaging and this had limitations on the number of messages that could be sent in 24 hours.

Although Twitter automatically gave a direct messaging system on a number of destination platforms including all mobile phones, tablets and PCs it was noted that a small number of learners did not use Twitter and had to sign up for accounts.

Given the limited messaging API provision by social networking providers and the range of providers that may or may not be used by learners, social networking may not be the best solution, messaging applications may offer a better system of communication, WhatsApp for example is “going to start inviting businesses onto the network” (Olson 2016) and so may offer a better messaging solution.

It is clear this investigation in isolation can only provide a small contribution to larger fields of study relating to the motivation of students to learn outside of a taught environment, what it does provide is an indication that students respond positively to automated prompting. It is hoped that the results will promote discussion and further study into other automated mechanisms that assist with learner engagement using prompting mechanisms. The tool developed for the investigation can be used as is in Moodle installations and also provides a framework for future Moodle plug-in developers both for research and for the development of plug-ins for general use. For course designers and tutors the plug-in provides a stable system for contacting small groups of students when they do not engage with important learning activities; it also provides a methodology for identifying if there is measurable improvement in engagement for their students. Should the plug-in be used in other academic institutions the system should at least provide helpful prompts for students to enable them to easily identify missed work before falling too far behind in their studies. VLE usage analysis software.

## 5.2 VLE USAGE ANALYSIS SOFTWARE

Research was conducted into the potential use of data held in the Moodle VLE at Lincoln College as a source of guidance for Lincoln College staff involved in the development of course materials for use on the Moodle VLE.

The overall aim of the research was to identify if course developers when presented with software tool representing data from a VLE such as Moodle would accept it as an effective tool to influence the design of a course.

The hypothesis is that the Moodle system contains student activity data that can be analysed and formatted in a suitable fashion that it would be accepted by course designers as a tool to aid them in evaluating the effectiveness of their course resources.

Specifically, the study intended to identify if the following questions could be answered.

- a) Is Moodle capable of providing tracking data about VLE resource utilisation from its internal databases in a suitable format to be further processed by the analysis software to be developed for this study?
- b) Can effective processing be performed on the data extracted from Moodle to provide meaningful information about resource utilisation?
- c) Can the result of the data processing be displayed in an easy to interpret format for a designer to use to make judgements on the level of engagement of students with resources?
- d) Is the information provided by the software useful to the designer to the point where the designer would consider data mining software a viable tool for evaluating VLE resources?

The research methodology included the development of a Microsoft Windows application to aid in the answering of questions a, b and c. The methodology selected to provide data to answer question d was through the collection of quantitative data using a video demonstration of the software and analysis of the results from an evaluative questionnaire.

#### **5.2.1 Is Moodle capable of providing the tracking data in a suitable format to be further processed?**

Before entering into the main software development activities for this study it was necessary to establish that accuracy of anecdotal evidence gathered from informal discussion that indicated that course designers do not effectively evaluate the utilisation of their course resources and that the data held in Moodle was in a format that could be extracted for use in the proposed analysis software.

The methodology used to confirm the anecdotal evidence regarding course design was through the development of a cross sectional survey, to establish:

- How developers use Moodle features
- How developers organise their resources
- How designers evaluate their course design
- Designers level of ability to gather data about student activity
- The importance of the materials to a student's success

The results from the survey indicate that within Lincoln College, Moodle courses are predominantly used as a structured store of reference materials with 48% including elements in their courses that students are expected to use for self-study and that 56% of the designers considered 40% or more of course resources as vital to a student completing a course.

This suggests that course designers do consider the structuring of online course resources and expect students to utilise them independently and consequently do need to evaluate the effectiveness of the resources.



When considering evaluating resource utilisation 68% of designers indicated they had at least an average level of ability in retrieving student activity data from Moodle but only 10% actually retrieved this data to review their courses with 43% using student feedback to evaluate course design and the remainder did not evaluate the course or guessed. These results indicate that although the majority of designers can retrieve the data there is little utilisation of it to evaluate the resources. This was clearly an opportunity for the study to have an impact by educating designers in the potential benefits of using data mining software for evaluation and improvement of the quality of VLE resources.

From the survey results there is a clear indication that Moodle courses and their utilisation by students is important to a student being successful in their studies; the results also suggest that although many designers are able to evaluate course design they are not using student usage data to confirm student feedback missing the opportunity for improvement proposed by Mohamad & Tasir (2013).

The tutors use of potentially biased student feedback and the low utilisation of Moodle data for evaluation suggests there may be a need for a tool to aid a designer to quantitatively evaluate the utilisation of their course resources. Findings from the literature search identified that both Mazza et al. (2012) and Romero et al. (2013) had been successful in extracting and using Moodle data linked to students. The studies identified the Moodle database as the source of the information and pre-processed the data prior to analysis, neither were clear in their papers as to the methods used to extract the data. Further investigation into the accessing of data from the Moodle database revealed that an existing "configurable reports plug-in" (Layva 2013) allowed custom SQL queries to be written by a course designer to extract data in the form of downloadable ODS or CSV files in for further processing.

The configurable reports plug-in provided a solution to retrieving data from the system to confirm that the data was available and investigation into the Moodle database schema resulted in an appropriate SQL query

being developed to extract the data shown in 8.2 Appendix B-Log extraction query with sample output. It was observed that the time taken for execution of queries from this plug-in was affected by a design flaw in the Moodle database schema that stored different activities in different tables that could only be resolved with multiple left joins giving poor query performance.

From both findings of the literature search and the practical investigation of the configurable reports plug-in confirm that it is possible to extract the data in an ODS (open document spreadsheet) format suitable for further processing. If further development were to take place it would be desirable to develop a specialised web API to link the data to the analysis system to make this stage efficient.

### **5.2.2 Can effective processing be performed on the data to provide meaningful information?**

The investigation into the ability to extract data from the Moodle VLE concluded that through the use of a customisable query in the VLE it was possible to create an ODS file containing student tracking data.

In section 3.2.2 Software development methods the development of the C# class shown in Figure 27-The ODS Analyse class was undertaken to process the ODS file, it was composed of an open source ODS reader and writer class (GemBox Software 2011) and methods to extract and process the data into list objects.

The use of C# and the ODS file proved to be a successful combination for the purpose of the study with development time being shorter than expected for this element. Data extraction code execution proved to be effective with processing times being typically < 1 second to read the ODS file and < 50 milliseconds to process the data that falls within the 10 second limit for “for keeping the user's attention focused...” (Nielsen 1993) and supported by Benson et al. (2014).

The result of the processing allowed for the linking of student information to their first and total accesses to study resources within course and the

ability to select this information by student grades and resource types. As a summative system for the evaluation this information provided a sound basis for the review of the course, establishing that it is possible to process the data into meaningful information.

### **5.2.3 Can the result of the processing be displayed in an easy to interpret format for a designer to use and is the information useful to the designer?**

In section 5.2.2 Can effective processing be performed on the data to provide meaningful information? It was identified that processing of the data could be achieved in a timely fashion that would link student accesses to resources within a course, confirming that data could be produced that would be meaningful to a designer.

The final stage of the research implementation investigated whether this data could be presented in a format that would be effective for a designer to use. The literature review identified that for previous investigations by Romero et al. (2013), Konstantinidis & Grafton (2013) and Mazza et al. (2012) that charts were the preferred method of data representation.

The developed solution formed part of a windows presentation foundation application that incorporated the previously created data analysis object to form windows desktop application capable of reading the data files, processing the data and displaying the results in a graphical form.

The graphical representation was an interactive bar graph (Figure 32, p. 61) representing the course resources providing a selection of sort orders and student and resources filters. User feedback at the design stage indicated that there was a need to include a simple quality measure in the form of the expected number of resource accesses based on the total number of students on the course being analysed.

The use of the online survey question shown in section 8.3.2 Moodle Log Analyser Survey, with a linked video presentation (Smith 2016) to provide quantitative data was selected as the methodology for identifying if the information was considered to be of use to course designers.

The responses from the survey question, "Would the tool be used in a designers' normal activity if it was provided to them?" revealed that 65% of the designers surveyed would use the software as part of their normal activity, 29% were unsure and 6% would not use the software. The results suggest that the data provided and its presentation was sufficiently clear to the designers for them to consider it as a tool for use in their normal course design activities. The overall low 6% rejection rate of the tool would indicate that there is a possibility of a higher proportion of acceptance if the 29% of respondents who answered "maybe" were shown further demonstrations of the tool to establish its "ease of use" (Sanchez & Hueros 2010). This further engagement with the participants was considered as a another step in the methodology but was rejected over concerns over the possibility of an "experimenter-expectancy effect" (Allen 2015) causing bias to the existing results. The analysis of the responses relating to the adoption of the software Figure 41-Chart of software adoption by designers who previously did/did not evaluate their courses also revealed that the level of adoption was consistent between the two groups offering some validity to the intention to improve adoption of data mining in evaluation of course resources. It suggests that designers who previously did not inform the enhancement their courses through data mining would now in many cases do so regardless of previously not having a strategy or relying purely on subjective feedback.

#### **5.2.4 VLE usage analysis software results summary**

The overall aim of the research was to identify if course developers when presented with software tool representing data from a VLE such as Moodle would accept is as an effective tool to influence the design of a course.

Through the analysis of the Moodle system and the subsequent development of the analysis software there is sufficient evidence to

suggest that data can be extracted and processed to deliver information to course designers. This finding is corroborated by the Moodle LMS development team stating in their learning analytics plan “Learning analytics are needed to inform teachers about students, to inform the staff that support teachers and to inform students themselves” (Moodle 2017). The results of the online surveys utilised during and post development have indicated there is a need for this type of software being developed and that from the survey data that a high proportion of designers would use it as part of their evaluation activities. The study has also highlighted that there is an issue around course developers in an educational environment not having sufficient skills to extract and process data from VLEs; for VLE producers in the wider educational community this indicates that there is a need to develop built-in tools that are not only effective but user friendly aiding course designers in understanding how effective their resources and activities are. The production of such tools would lead to improvement in the quality of the learning experience for course users and courses with increasingly engaged learners.

## 6 CONCLUSIONS

---

The memorable quotes “*Good habits formed at youth make all the difference*” made by the Greek philosopher Aristotle and “*I never teach my pupils. I only attempt to provide the conditions in which they can learn*” by Albert Einstein inspired the aim of improving the level of engagement with students with Moodle Lincoln Colleges VLE instilling a good habit of learning when away from college and identifying online course resources and activities that meet their needs.

This intent led to the research question “How might the development of data mining and log analysis systems for the Moodle virtual learning environment (VLE), improve computer science students' course engagement and encourage course designers' future engagement with data analysis methods for the evaluation of course resources?”.

The literature search revealed that there had been a number of investigations into the extraction data from Moodle resulting in software implementations such as “GISMO” (Mazza & Milani 2004) , “MOCLog” (Mazza et al. 2012), investigations by Liu et al. (2016) and spreadsheet solutions Konstantinidis & Grafton (2013). These implementations focused on the provision of data to allow a course deliverer to interpret what interactions are being made during a course for the purpose of establishing whether to contact a student regarding their activity level. All three investigations are reliant on there being direct interactions with the VLE by the course deliverer and the student accessing the VLE in the case of MOCLog.

The literature search also revealed that although Mohamad & Tasir in Educational Data Mining: A Review noted that “...perhaps we can shift our focus from the e-learning, towards the use of social networking tools...” (2013), most social networks restrict electronic messages to their internal messaging applications and systems. Twitter however did provide an API that allowed direct messaging from an external source which gave some validity to Mohamad & Tasir's recommendations.

The overall aim of this research is to examine if systems based on data mining can improve the quality of VLE based learning and student engagement. The two main objectives established to meet this aim were to create VLE based data mining system that could improve engagement with a VLE course and to provide a software application for data mining VLE course utilisation to encourage acceptance and future engagement with the technology as a viable form of evaluation in a test-and-improve course development cycle. This was achieved through the development of two software artefacts that provide automated prompting to encourage students to engage with a course (MooTwit) and the provision of information about the utilisation of courses (MooLog).

The development of the MooTwit and MooLog software formed a large part of the methodology and through their development a number of insights were gained into the practicalities of working with VLE systems (specifically Moodle). Both system designs focused on ease of use being a priority for user acceptance as the literature search highlighted it as a factor influencing the acceptance of Moodle in general. MooTwit approached development through integrating with the VLE directly in the form of a plug-in that allowed the tutor to set expected completion dates for activities and resources within a course, which was subsequently monitored against student activity on the system prompting and congratulating them appropriately. MooLog was presented as a desktop application that processed exported student activity data from the VLE linking the activity to the course elements; the results were presented graphically to enable effective interpretation of the data mined results.

A significant finding from the development process is that the use of offline processing used MooLog is less time effective for the software user than the VLE plug-in systems suggesting that systems used in MooTwit would be the effective approach for future developments and may further encourage user acceptance of the technology extending the study's aim of increasing engagement with data mining for improving course quality.

## 6.1 CONCLUSIONS DRAWN FROM MOOTWIT

All studies previously undertaken have focused on the methods of analysing and displaying information but have not investigated if this could be used as a positive effect on the student. The primary goal of the MooTwit development was to extend the data mining process in to a system that improved student engagement with VLE courses by analysing their accesses and automatically prompting them via social networking backed up with email when they were falling behind in accessing activities on the VLE.

The hypothesis was that the experimental group (students receiving automated prompting) would access missed VLE course materials sooner than the control group. The results of the study support this hypothesis suggesting that electronic tracking and prompting of students who miss activities does have an a small to medium effect on the timeliness of their engagement with the VLE activities. Specifically, the results suggest that when students receive a Twitter direct message to engage with missed material, they complete missed activities closer to required completion date compared to the unprompted students. In the context of the initial aim of the research of improving engagement of students with the activities, clearly there has been a demonstrable change in behaviour by students in the experimental group. There is a good indication that the adoption of the system would be beneficial, increasing student utilisation of course resources at the appropriate point in their studies, enhancing the quality of their learning experience. There are some limitations to the extent of this success in that these results may not be reflected in the context of the wider learner community; this experiment focused on a specific group of students, the success shown here may not yield the same result with different learner demographic. This limitation provides scope for extending the investigation into a larger domain to establish if the findings can be completely generalised by varying such parameters such as the level of technological ability, initial engagement level of the students with the VLE, the age of the participants or course subject studied.



In the preliminary stages of reviewing the results of the research there was a concern that the use of prompting in the system may have demotivated the students or that the effect of prompting would be less effective over time. The initial statistical data analysis addressed demotivation in terms of accepting as a whole the statistical outcome was positive; it did not however consider the use of the prompting over time. The students may have been receptive to the system at the start of the experiment, but less so by the end of their studies. The use of a t-test and Cohens D effect level did not provide a temporal measure to allow this level of analysis. The study addressed this limitation by analysing the student behaviour over the duration of the experiment identifying that the experimental group showed a positive trend in the mean timeliness of access to course resources over the 15 weeks of the experiment indicating that repeated prompting does have a positive cumulative effect over time. This result in combination with the findings of Miltenberger (2011) on the use of prompting and transfer of stimulus control would suggest the automated prompting has a similar effect to prompting by a course deliverer and that there is a transfer of stimulus of control over time. As with most studies of this scale there is opportunity to extend this investigation both in terms of the refinement of the methods used to prompt the participants and to further study participant behaviour (*good habits*) post experiment, given that the intent of the process was to improve engagement it is especially important to confirm if the effects were temporary. A longer study may also highlight if there would be an element of apathy in responding to the prompts as they became the accepted norm.

Overall this investigation into improving student engagement provides evidence to support the case for data mining of VLE data stores to aid in establishing good study habits in learners. While social networking as a form of communication was used successfully as part of this small study, the restrictions in the majority of social networking infrastructures limit the implementation and scalability of the process in its current form. There are also concerns over the volatility of the popularity of social networks; while at the time of writing Twitter still has popularity now, Instagram for example has a growing user base. On the basis of the varying level of adoption of

different social networking sites it is suggested that future areas for study could include the use of other low cost and low latency communication systems such as Skype or Google Firebase Cloud Messaging for communication of prompts to the learners as an alternative to social networking effectively eliminating the issue of which social network to use and providing a dedicated path of communication between the system and the student.

## **6.2 CONCLUSIONS DRAWN FROM MOOLOG**

With a growing reliance on VLE systems within the education sector the impact of students not engaging with online learning activities will become increasingly significant; consequently, it is imperative that the area of motivating learners to engage with these systems is investigated and solutions found. The second element of the research undertaken (MooLog) developed data mining software identifying the VLE resources most utilised against student overall performance on a course to encourage future acceptance of data mining as viable resource evaluation system to improve the quality of VLE resources provided to learners and consequently improve learner engagement with the courses. The investigation used surveys to collect data from a group of 50 course designers who were actively engaged with the Lincoln colleges Moodle VLE. The survey questions focused on how the designers currently use Moodle and surveyed them on the features provided by MooLog to identify if the system would enhance their decision making when evaluating course designs and as a consequence of this was there a change in attitude to the adoption of data mining for "test and improve" course development over existing methods.

The study revealed that while most course designers considered how the materials might be accessed by students, virtually all designers used the VLE as a repository for course materials. The majority provided materials in a structured format for learners and half of them expected the courses to be used for self-study. Many considered their VLE based resources as being vital to student success. The findings would indicate that VLE courses form a significant part of the learning experience and the utilisation of

materials and activities provided for students needed to be evaluated to ensure that what is provided fits with learner needs. The existing methods the study group used for evaluating course relied on student feedback or methods not being used at all, in-fact only 10% of designers used data as a form of evaluation. There is evidence to suggest that there is either a lack of awareness of the benefits of data mining technologies for evaluation or a lack of data processing ability within the study group; a third of which acknowledged that they were below average or were unable at retrieve information and less than half evaluated themselves as highly skilled in retrieving information. The lack of utilisation of data in making decisions about resource effectiveness and the high proportion of designers who would struggle to extract and analyse data supported the premise that the development of a data mining application could encourage continuous empirical resource evaluation to identify possible areas of failure in the course design highlighting what needs improving. While the results from the investigation identify there is a need to evaluate, they do not by themselves identify if data mining would be accepted by the designers as a valid mechanism or that they would consider themselves as technically capable of performing the analysis in the future.

As part of the investigation MooLog was written to provide a user friendly demonstrable software application to identify if the designers would accept the need for resource evaluation using a data mining system. Designers were asked to evaluate the potential usefulness of the information provided by the software and to establish if they would utilise the data mining software as part of their evaluative processes in the future. The use of fully functional software combined with electronic surveys proved to be an effective method of working with the study group allowing participant clear visualisation of the simple processes for data analysis and a quick turnaround of responses in a quantitative format.

The results of the study established there was evidence that the designers' acceptance of the potential of data mining was significantly increased with the original use of data mining being only 10% compared with the 65% of designers stating they would use the data mining software after the

demonstration, the results suggest that although many designers currently do not use data analysis as a method of informing course improvement when offered an easy to use system for viewing the data the majority would value the information provided and would adopt it as part of their normal activity. It is important to note that worryingly there were a significant number of participants that were not using any form of evaluation for course iterations. This study has encouraged just over half of them to adopt the system and over a quarter to consider it as a viable option in-line with the original intention of the study of improving the quality of VLE linked courses and to increase future acceptance of data mining for evaluation of resources. Providing the MooLog application has resulted in change of attitude to data mining in a large proportion of the designers suggesting that the barrier to designers using data mining to evaluate their courses is a combination of not understanding the value of the process and the lack of simple tools such as MooLog to reduce the complexity of analysing data into meaningful results. The designers clearly saw the worth of the information when surveyed and out of those designers who indicated that they would utilise MooLog at least half still wanted training; this response would indicate that process is still a barrier. Further simplification of the VLE data extraction process to provide the data without involving the user could potentially further increase the adoption of the system. These findings support the case for development and utilisation of tools to aid in VLE course evaluation; confirming the assumption in the works of Mazza et al. (2012); Romero et al. (2013); Sanchez & Hueros (2010) and Zhang & Almeroth (2010) that data mining tools would be used if provided. Since completion of the study the Moodle LMS designers have now started to move forward with this through Project Inspire that aims to "...develop open source, transparent next-generation learning analytics that go beyond simple descriptive analytics to provide predictions of learner success, and ultimately diagnosis and prescriptions (advisements) to learners and teachers..." (Dalton 2017).

Given the small scale of the investigation there are opportunities for further research to see if the high acceptance level shown by the course designers at Lincoln College can be generalised. There is clearly further

work to be done in relation to data mining of educational systems both in providing the facilities to a broader spectrum of VLE's such as Canvas, Blackboard and Google Classroom. Possibly the actual interface to the mining process could be significantly enhanced, the use of voice recognition and natural language is becoming increasingly prevalent as assistive technologies in systems such as IBM, Google, Amazon and Apple technologies, would systems such as this aid or hinder adoption?

### **6.3 CLOSING THOUGHTS**

Overall the subjects of improving the quality of student experience and engagement with electronic resources explored in this study have opened many further lines of enquiry about technologies role in learning and the consequences of not engaging with them as a learner and how to evaluate the quality of what is provided to the learner in the first place. There is a compelling argument for encouraging student engagement with VLE based resources from the findings from the MooTwit experiment and for the evaluation of the utilisation of what is provided to a student through software such as MooLog. Data mining has a role in both purposes and given that data analytics and mining are tools that are prevalent in many internet and customers related industries, these tools could ultimately have a significant positive impact on the quality of learning and abilities of the students that leave education. Alongside the VLE content provided by educational institutions the Internet and world wide web has provided a gateway to an almost endless collection of resources for learning, many of which are either poorly structured, badly developed or inaccurate but are frequently visited by learners in preference to the content provided by educational institutions. The biggest unanswered question from this study is why do students opt for information from a less reliable source or spend time looking for information that they would have found if they had engaged with the system provided to them?

## 7 REFERENCE LIST

---

- 1&1 Internet Inc., 2015. Send E-mails Using PHP. Available at: <https://help.1and1.com/hosting-c37630/scripts-and-programming-languages-c85099/php-c37728/send-e-mails-using-php-a657019.html> [Accessed January 10, 2015].
- Affairs, A.S. for P., 2013. Paper Prototypes Work as Well as Software Prototypes.
- Ahmad, A. & Talha, M., 2002. A measurement based comparative evaluation of effectiveness of object-oriented versus conventional procedural programming techniques and languages. *Ninth Asia-Pacific Software Engineering Conference, 2002.*, pp.517–526. Available at: <http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=1183072>.
- Allen, M., 2015. The experimenter expectancy effect : an inevitable component of school science ? *Research in Education*, 94(1), pp.13–29.
- Barnham, C., 2015. Quantitative and qualitative research: perceptual foundations. *International Journal of Market Research*, 57(6), pp.837–854.
- Benson, C., Bryan, C. & Nickell, S., 2014. Acceptable Response Times. *GNOME Human Interface Guidelines 2.2.3*. Available at: <https://developer.gnome.org/hig-book/unstable/feedback-response-times.html.en> [Accessed November 27, 2016].
- Blair, J., Czaja, R.F. & Blair, E.A., 2013. *Designing Surveys: A Guide to Decisions and Procedures*, SAGE Publications. Available at: <https://books.google.co.uk/books?id=laQgAQAAQBAJ>.
- Bloh, C., 2008. Assessing transfer of stimulus control procedures across learners with autism. *The Analysis of verbal behavior*, 24, pp.87–101. Available at: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2779927&tool=pmcentrez&rendertype=abstract>.

- Boehm, B.W. et al., 1987. A Spiral Model of Software Development and Enhancement. *Computer*, 21(May), pp.61–72. Available at: [http://ieeexplore.ieee.org.ezproxy.herts.ac.uk/ielx1/2/6/00000059.pdf?tp=&arnumber=59&isnumber=6%5Cnhttp://ieeexplore.ieee.org.ezproxy.herts.ac.uk/xpls/abs\\_all.jsp?arnumber=59%5Cnhttp://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=59](http://ieeexplore.ieee.org.ezproxy.herts.ac.uk/ielx1/2/6/00000059.pdf?tp=&arnumber=59&isnumber=6%5Cnhttp://ieeexplore.ieee.org.ezproxy.herts.ac.uk/xpls/abs_all.jsp?arnumber=59%5Cnhttp://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=59).
- Booch, G., 1999. *Object-Oriented Analysis and Design*, Redwood City: Benjamin Cummings.
- Center for Evidence-Based Management, 2014. What are the levels of evidence? Available at: <https://www.cebma.org/faq/what-are-the-levels-of-evidence/> [Accessed May 2, 2014].
- CheckMarket, 2015. Sample size calculator. Available at: <https://www.checkmarket.com/sample-size-calculator/> [Accessed October 7, 2015].
- Cohen, J., 1992. Quantitative Methods in Psychology. , 112(1), pp.155–159.
- Currie, K., 2016. Scalable and Repeatable Development with a Personalised Touch. In *MoodleMoot Ireland UK 2016*. Available at: [https://mootieuk.moodlemoot.org/pluginfile.php/1800/mod\\_data/content/1964/MoodleMoot-PK.pptx](https://mootieuk.moodlemoot.org/pluginfile.php/1800/mod_data/content/1964/MoodleMoot-PK.pptx).
- Curtis, G. & Cobham, D.P., 2008. *Business Information Systems: Analysis, Design & Practice*., Pearson Education M.U.A. Available at: <https://www.dawsonera.com:443/abstract/9781408212622>.
- Dale, C., 2007. A Wolf in Sheep's Clothing? An Analysis of Student Engagement with Virtual Learning Environments. *The Journal of Hospitality Leisure Sport and Tourism*, 6(2), pp.100–108. Available at: [http://www.heacademy.ac.uk/assets/hlst/documents/johlste/vol6n02/156\\_dale\\_vol6no2.pdf](http://www.heacademy.ac.uk/assets/hlst/documents/johlste/vol6n02/156_dale_vol6no2.pdf) [Accessed April 2, 2014].
- Dalton, E., 2017. Project Inspire. Available at:

<https://moodle.org/course/view.php?id=17233> [Accessed March 5, 2017].

Davies, A. & Fisk, H., 2006. *MySQL Clustering*, MySQL Press.

Davis, F., Bagozzi, R. & Warshaw, P., 1989. User acceptance of computer technology: A comparison of two theoretical models. *Management Science*, 35(8), pp.982–1003. Available at: <http://mansci.journal.informs.org/cgi/doi/10.1287/mnsc.35.8.982>.

Dawson, P., 2014. A learning analytics block to identify students at risk of disengaging with their studies – \$50k, NetSpot Innovation Fund, 2012. Available at: <http://philldawson.com/moodle-learning-analytics/> [Accessed March 3, 2014].

Drachsler, H. & Greller, W., 2016. Privacy and Analytics – it ' s a DELICATE Issue A Checklist for Trusted Learning Analytics. *6th Conference on Learning Analytics and Knowledge*, p.10.

Facebook, 2013. Requests Dialog. Available at: <https://developers.facebook.com/docs/reference/dialogs/requests/>.

Fielding, R., 2000. *Architectural Styles and the Design of Network-based Software Architectures*. University of California.

GemBox Software, 2011. How to read and write ODF/ODS files (OpenDocument Spreadsheets). Available at: <http://www.gemboxsoftware.com/spreadsheet/articles/read-write-ods-odf-net>.

Greenhow, C., Robelia, B. & Hughes, J.E., 2009. Learning, Teaching, and Scholarship in a Digital Age: Web 2.0 and Classroom Research: What Path Should We Take Now? *Educational Researcher*, 38(4), pp.246–259. Available at: <http://edr.sagepub.com/cgi/doi/10.3102/0013189X09336671>.

Halpin, B.P.A., 2016. Using Twitter in a Nonscience Major. , 45(6), pp.71–78.



- Hampel, R. & Pleines, C., 2013. Fostering student interaction and engagement in a virtual learning environment: An investigation into activity design and implementation. *CALICO Journal*, pp.342–370. Available at: <http://journals.sfu.ca/CALICO/index.php/calico/article/view/975> [Accessed April 2, 2014].
- Hughes, G., 2005. *Learning-to-learn online: fostering student engagement with online pedagogies* in P. Hartley, A. Woods, & M. Pill, eds., Taylor & Francis. Available at: [https://books.google.co.uk/books?id=g9F\\_AgAAQBAJ](https://books.google.co.uk/books?id=g9F_AgAAQBAJ).
- Instagram, 2014. Instagram API Resources. Available at: <https://apigee.com/resources/instagram>.
- Jaans, 2012. DotNetZip - Zip and Unzip in C#, VB, any .NET language. Available at: <http://dotnetzip.codeplex.com/>.
- Jacobsen, I. et al., 1999. *Object-Oriented Software Engineering*, Wokingham: Addison-Wesley.
- JISC, 2016. Technology and tools for online learning | Jisc. Available at: <https://www.jisc.ac.uk/full-guide/technology-and-tools-for-online-learning> [Accessed May 6, 2017].
- Junco, R., Heiberger, G. & Loken, E., 2011. The effect of Twitter on college student engagement and grades. *Journal of Computer Assisted Learning*, 27(2), pp.119–132.
- Kaur, P., Singh, M. & Josan, G.S., 2015. Classification and Prediction Based Data Mining Algorithms to Predict Slow Learners in Education Sector. *Procedia Computer Science*, 57, pp.500–508. Available at: <http://dx.doi.org/10.1016/j.procs.2015.07.372>.
- Kelley, K. et al., 2003. Good practice in the conduct and reporting of survey research. *International Journal for Quality in Health Care*, 15(3), pp.261–266.
- Khan, M.A., Parveen, A. & Sadiq, M., 2014. A method for the selection of software

- development life cycle models using analytic hierarchy process. 2014 *International Conference on Issues & Challenges in Intelligent Computing Techniques (ICICT)*, p.534. Available at: <http://proxy.library.lincoln.ac.uk/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=edb&AN=96148246&site=eds-live>.
- Knight, C.G. & Kaye, L.K., 2014. "To tweet or not to tweet?" A comparison of academics' and students' usage of Twitter in academic contexts. *Innovations in Education and Teaching International*, 3297(April 2015), pp.1–11. Available at: <http://www.tandfonline.com/doi/abs/10.1080/14703297.2014.928229%5Cnhttp://www.scopus.com/inward/record.url?eid=2-s2.0-84902938938&partnerID=tZOtx3y1>.
- Konstantinidis, A. & Grafton, C., 2013. Using Excel Macros to Analyse Moodle Logs. *Research.Moodle.Net*, pp.4–6. Available at: [http://research.moodle.net/pluginfile.php/333/mod\\_data/content/1233/Using Excel Macros to Analyse Moodle Logs.pdf](http://research.moodle.net/pluginfile.php/333/mod_data/content/1233/Using%20Excel%20Macros%20to%20Analyse%20Moodle%20Logs.pdf).
- Kuh, G.D. et al., 2011. *Student Success in College: Creating Conditions That Matter*, Wiley. Available at: <https://books.google.co.uk/books?id=9KnzFmNYuBMC>.
- Kumar, S., Morstatter, F. & Liu, H., 2013. *Twitter Data Analytics*, Springer New York. Available at: <https://books.google.co.uk/books?id=Q164BAAQBAJ>.
- Layva, J., 2013. Configurable reports. Available at: [https://docs.moodle.org/27/en/Configurable\\_reports](https://docs.moodle.org/27/en/Configurable_reports) [Accessed February 7, 2014].
- Lazar, J., Feng, J.H.J. & Hochheiser, H., 2010. *Research methods in human-computer interaction*, Available at: [http://books.google.com/books?hl=en&lr=&id=H\\_r6prUFpc4C&oi=fnd&pg=PT17&dq=RESEARCH+METHODS+IN+HUMAN-COMPUTER+INTERACTION&ots=xX61FobbFG&sig=coOHlvUNdv27XIQHjMB-eaidQBM](http://books.google.com/books?hl=en&lr=&id=H_r6prUFpc4C&oi=fnd&pg=PT17&dq=RESEARCH+METHODS+IN+HUMAN-COMPUTER+INTERACTION&ots=xX61FobbFG&sig=coOHlvUNdv27XIQHjMB-eaidQBM).

- Liu, D.Y.T. et al., 2016. Knowledge Acquisition for Learning Analytics: Comparing Teacher-Derived, Algorithm-Derived, and Hybrid Models in the Moodle Engagement Analytics Plugin. In *Pacific Rim Knowledge Acquisition Workshop*. pp. 183–197.
- Maltby, A. & Mackie, S., 2009. Virtual learning environments – help or hindrance for the “ disengaged ” student ? *ALT-J, Research in Learning Technology*, 17(March), pp.49–62.
- Mazza, R. et al., 2012. MOCLog – Monitoring Online Courses with log data S. Retalis & M. Dougiamas, eds. *1st Moodle Research Conference*, pp.132–139. Available at: <http://research.moodle.net/mod/data/view.php?d=3&rid=17>.
- Mazza, R. & Milani, C., 2004. GISMO: a Graphical Interactive Student Monitoring Tool for Course Management Systems. In *T.E.L.'04 Technology Enhanced Learning '04 International Conference. Milan*. pp. 18–19.
- Microsoft, 2015. *CollectionViewSource* class. Available at: <https://msdn.microsoft.com/library/windows/apps/br209833> [Accessed September 10, 2015].
- Microsoft, 2010. *Introduction to WPF*. Available at: [https://msdn.microsoft.com/en-us/library/aa970268\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/aa970268(v=vs.100).aspx) [Accessed August 3, 2016].
- Microsoft, 2012. *The MVVM Pattern*. Available at: <https://msdn.microsoft.com/en-gb/library/hh848246.aspx> [Accessed June 15, 2015].
- Miltenberger, R., 2011. *Behavior Modification: Principles and Procedures*, Cengage Learning.
- Mohamad, S.K. & Tasir, Z., 2013. Educational Data Mining: A Review. *Procedia - Social and Behavioral Sciences*, 97, pp.320–324. Available at: <http://linkinghub.elsevier.com/retrieve/pii/S1877042813036859> [Accessed March 27, 2014].

Mohammed, N., Munassar, A. & Govardhan, A., 2010. A Comparison Between Five Models Of Software Engineering. *International Journal of Computer Science*, 7(5), pp.94–101.

Mono Project, 2016. Cross platform, open source .NET framework. Available at: <http://www.mono-project.com/> [Accessed September 30, 2016].

Moodle, 2017. Why is a Learning Analytics system needed? Available at: [https://docs.moodle.org/dev/Learning\\_Analytics\\_Specification#Why\\_is\\_a\\_Learning\\_Analytics\\_system\\_needed.3F](https://docs.moodle.org/dev/Learning_Analytics_Specification#Why_is_a_Learning_Analytics_system_needed.3F) [Accessed March 5, 2017].

Moodle Docs, 2015a. Data manipulation API. Available at: [https://docs.moodle.org/dev/Data\\_manipulation\\_API](https://docs.moodle.org/dev/Data_manipulation_API) [Accessed July 14, 2016].

Moodle Docs, 2011. File:Moodle-contexts-1.8.png. Available at: <https://docs.moodle.org/dev/File:Moodle-contexts-1.8.png> [Accessed August 1, 2016].

Moodle Docs, 2016a. Form API - MoodleDocs. Available at: [https://docs.moodle.org/dev/Form\\_API](https://docs.moodle.org/dev/Form_API) [Accessed July 12, 2016].

Moodle Docs, 2013. Logs - MoodleDocs. Available at: <http://docs.moodle.org/26/en/Logs> [Accessed April 4, 2014].

Moodle Docs, 2016b. Plugin types - MoodleDocs. Available at: [https://docs.moodle.org/dev/Plugin\\_types](https://docs.moodle.org/dev/Plugin_types) [Accessed June 1, 2016].

Moodle Docs, 2015b. Roles - MoodleDocs. Available at: [https://docs.moodle.org/dev/Roles#The\\_new\\_roles\\_and\\_capability\\_system](https://docs.moodle.org/dev/Roles#The_new_roles_and_capability_system) [Accessed May 11, 2015].

Moodle Docs, 2016c. Task API - MoodleDocs. Available at: [https://docs.moodle.org/dev/Task\\_API#Scheduled\\_tasks](https://docs.moodle.org/dev/Task_API#Scheduled_tasks).

Nielsen, J., 1993. *Usability Engineering*, Morgan Kaufmann.

- OASIS, 2007. Key Features of OpenDocument Format. Available at: <http://www.opendocumentformat.org/features/> [Accessed October 9, 2014].
- Ofsted & The Office for Standards in Education, 2009. Virtual learning environments: an evaluation of their development in a sample of educational settings. *Alexandra House*, (January), pp.1–28. Available at: <http://www.ofsted.gov.uk>.
- Olson, P., 2016. WhatsApp Is Finally Inviting Businesses Onto Its Massive Network This Year. *Forbes*. Available at: <http://www.forbes.com/sites/parmyolson/2016/01/18/whatsapp-businesses-free-1-billion/#54a64a46b6bd> [Accessed October 30, 2016].
- Ortigosa, A. & Carro, R.M., 2003. The Continuous Empirical Evaluation Approach: Evaluating Adaptive Web-based Courses. In *Proceedings of the 9th International Conference on User Modeling*. UM'03. Berlin, Heidelberg: Springer-Verlag, pp. 163–167. Available at: <http://dl.acm.org/citation.cfm?id=1759957.1759985>.
- Petticrew, M. & Roberts, H., 2003. Evidence, hierarchies, and typologies: horses for courses. *Journal of Epidemiology and Community Health*, 57(7), pp.527–529.
- Romero, C., Castro, C. & Ventura, S., 2013. A Moodle Block for Selecting , Visualizing and Mining Students ' Usage Data.
- Romero, C., Ventura, S. & García, E., 2008. Data mining in course management systems: Moodle case study and tutorial. *Computers and Education*, 51(1), pp.368–384. Available at: <http://linkinghub.elsevier.com/retrieve/pii/S0360131507000590> [Accessed March 22, 2014].
- Salkind, N., 2014. *Statistics for People Who Hate Statistics*, SAGE Publications Ltd.
- Sanchez, R.A. & Hueros, A.D., 2010. Motivational factors that influence the

- acceptance of Moodle using TAM. *Computers in Human Behavior*, 26(6), pp.1632–1640.
- Savev, S. & Bailey, P., Experiences using F # for developing analysis scripts and tools over search engine query log data. , pp.1–16.
- Sekelsenmat, F., 2015. Multiplatform Programming Guide. Available at: [http://wiki.freepascal.org/index.php?title=Multiplatform\\_Programming\\_Guide&oldid=105856](http://wiki.freepascal.org/index.php?title=Multiplatform_Programming_Guide&oldid=105856) [Accessed December 15, 2016].
- Simon, T. & Rosch, S., 2015. Comparing the object-oriented extension with the classical IEC 61131-3 regarding reusability and understandability - A case study. *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, 2015–Octob.
- Smith, S., 2016. Questionnaire Demo. Available at: [https://www.youtube.com/watch?v=3HZ\\_FnwPP4A](https://www.youtube.com/watch?v=3HZ_FnwPP4A).
- Sommerville, I., 2015. *Software Engineering* 10th ed., Pearson. Available at: <http://iansommerville.com/software-engineering-book/>.
- Spence, R., 2001. *Information Visualization*, Addison-Wesley. Available at: <https://books.google.co.uk/books?id=gYQoAQAAMAAJ>.
- Student Learning Development, 2009. *Bar charts*, Leicester: University of Leicester.
- Suh, Y. & Snodgrass, R.T., 2016. Author ' s Accepted Manuscript An Empirical Study of Transaction Throughput Thrashing. *Information Systems*. Available at: <http://dx.doi.org/10.1016/j.is.2016.12.004>.
- Szolnoki, G. & Hoffmann, D., 2013. Online, face-to-face and telephone surveys - Comparing different sampling methods in wine consumer research. *Wine Economics and Policy*, 2(2), pp.57–66. Available at: <http://dx.doi.org/10.1016/j.wep.2013.10.001>.
- Tewari, A., 2013. Why You Should Use MVVM for Small Apps. *Applied Information*

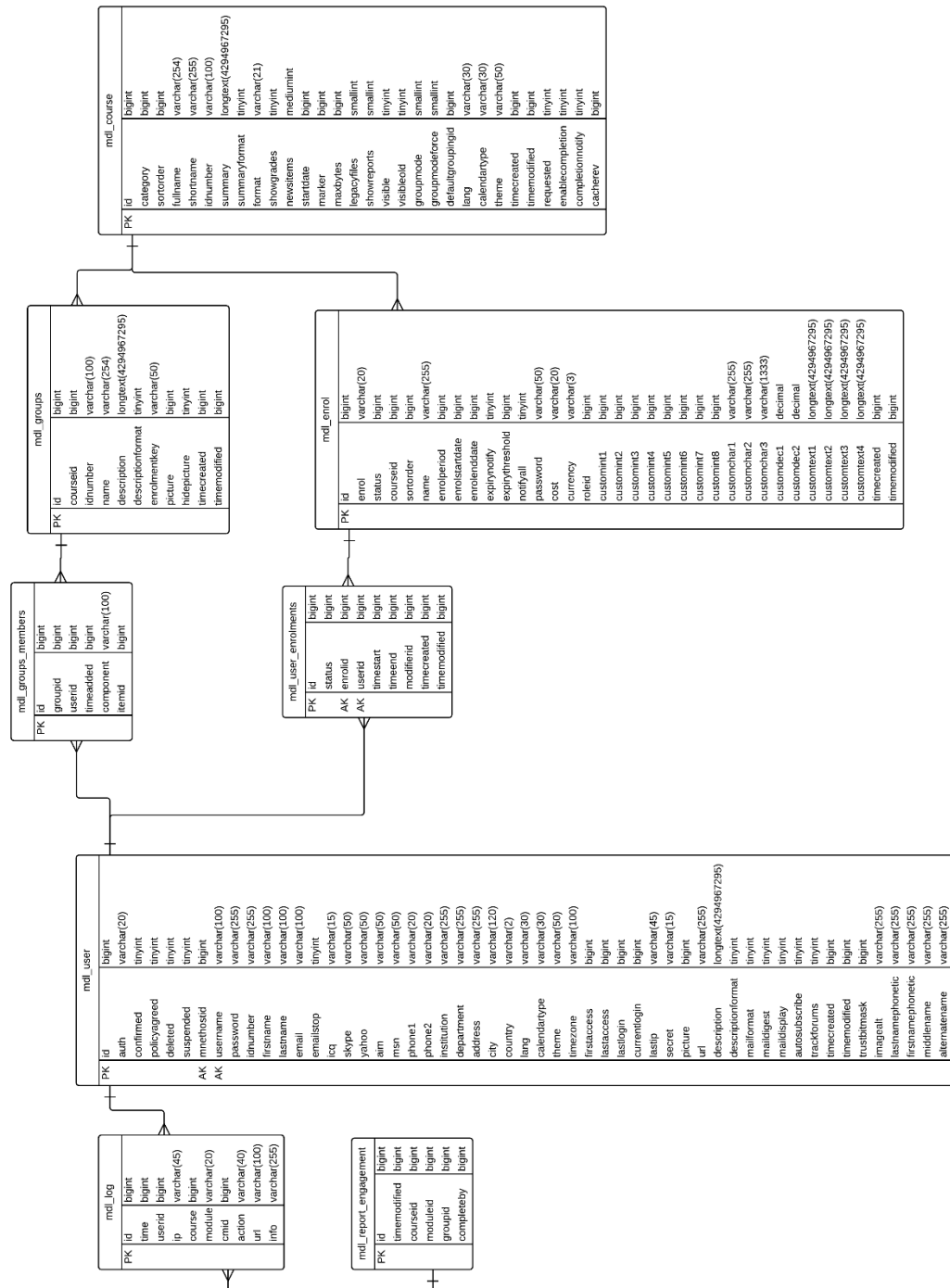
- Sciences. Available at: <https://blog.appliedis.com/2013/05/08/why-you-should-use-mvvm-for-small-apps/> [Accessed August 6, 2015].
- The PHP Group, 2016. PHP: General Information. Available at: <http://php.net/manual/en/faq.general.php> [Accessed July 26, 2016].
- Twitter, 2014. Twitter Help Center | Twitter limits (API, updates, and following). Available at: <https://support.twitter.com/articles/15364-about-twitter-limits-update-api-dm-and-following> [Accessed June 4, 2014].
- Uvalic-Trumbic, S. & Daniel, S.J., 2013. A Guide to Quality in Online Learning. *Academic Partnerships*.
- Walker, R., Voce, J. and Ahmed, J., 2014. Survey of technology enhanced learning for higher education in the UK. , p.10. Available at: <http://www.ucisa.ac.uk/groups/ssg/surveys.aspx>.
- Weller, M., 2007. *Virtual Learning Environments: Using, Choosing and Developing Your VLE*, Taylor & Francis. Available at: <https://books.google.co.uk/books?id=Cht9AgAAQBAJ>.
- Williams, A., 2016. TwitterOAuth. Available at: <https://twitteroauth.com/> [Accessed July 14, 2016].
- Wu, D. et al., 2015. A metrics-based comparative study on object-oriented programming languages. *Proceedings of 27th International Conference on Software Engineering and Knowledge Engineering*, pp.272–277.
- Yen, D. et al., 2015. An enhanced learning analytics plugin for Moodle: student engagement and personalised intervention. *Globally connected, digitally enabled. Proceedings ascilite 2015*, pp.180–189.
- Zhang, H. & Almeroth, K., 2010. Moodog: Tracking Student Activity in Online Course Management Systems. *Journal of Interactive Learning Research*, 21(3), pp.407–429. Available at: <http://www.editlib.org/p/32307/%5Chttp://www.editlib.org/p/32307>.





## 8 APPENDICES

## 8.1 APPENDIX A—MOODLE ENGAGEMENT PLUG-IN ERD



## 8.2 APPENDIX B-LOG EXTRACTION QUERY WITH SAMPLE OUTPUT

```

SELECT log.id, ctx.instanceid, ctx.contextlevel, cm.course,
       cm.module, cm.instance, mdl.name AS type,

CASE
  WHEN mf.name IS NOT NULL THEN mf.name
  WHEN mb.name IS NOT NULL THEN mb.name
  WHEN mr.name IS NOT NULL THEN mr.name
  WHEN mu.name IS NOT NULL THEN mu.name
  WHEN mq.name IS NOT NULL THEN mq.name
  WHEN mp.name IS NOT NULL THEN mp.name
  WHEN ml.name IS NOT NULL THEN ml.name
  WHEN mc.name IS NOT NULL THEN mc.name
  WHEN ma.name IS NOT NULL THEN ma.name
  WHEN mw.name IS NOT NULL THEN mw.name
  WHEN mch.name IS NOT NULL THEN mch.name

ELSE NULL
END AS description,
log.time, log.action, log.url, log.userid, usr.firstname,
usr.lastname, usr.username, crs.fullname, csec.section,
csec.name, csec.sequence

FROM prefix_course AS crs

INNER JOIN prefix_course_modules AS cm ON crs.id = cm.course
INNER JOIN prefix_course_sections AS csec ON cm.section = csec.id

INNER JOIN prefix_context AS ctx ON ctx.contextlevel = 70
AND ctx.instanceid = cm.id

INNER JOIN prefix_modules AS mdl ON cm.module = mdl.id
AND mdl.name != 'label' AND mdl.name != 'glossary'

INNER JOIN prefix_log AS log ON cm.id = log.cmid
INNER JOIN prefix_user AS usr ON log.userid = usr.id

LEFT JOIN prefix_forum AS mf ON mdl.name = 'forum' AND cm.instance = mf.id
LEFT JOIN prefix_book AS mb ON mdl.name = 'book' AND cm.instance = mb.id
LEFT JOIN prefix_resource AS mr ON mdl.name = 'resource' AND cm.instance = mr.id
LEFT JOIN prefix_url AS mu ON mdl.name = 'url' AND cm.instance = mu.id
LEFT JOIN prefix_quiz AS mq ON mdl.name = 'quiz' AND cm.instance = mq.id
LEFT JOIN prefix_page AS mp ON mdl.name = 'page' AND cm.instance = mp.id
LEFT JOIN prefix_lesson AS ml ON mdl.name = 'lesson' AND cm.instance = ml.id
LEFT JOIN prefix_choice AS mc ON mdl.name = 'choice' AND cm.instance = mc.id
LEFT JOIN prefix_assign AS ma ON mdl.name = 'assign' AND cm.instance = ma.id
LEFT JOIN prefix_workshop AS mw ON mdl.name = 'workshop'
AND cm.instance = mw.id
LEFT JOIN prefix_checklist AS mch ON mdl.name = 'checklist'
AND cm.instance = mch.id

WHERE crs.id = %%COURSEID%% AND log.cmid > 0
%%FILTER_STARTTIME:time:>%% %%FILTER_ENDTIME:time:<%% ORDER BY time

```

id	instanceid	course	module	instance	type	description	time	action	url	userid	firstname	lastname	username	fullname	section	name	sequence
15387	204	70	6	15	5	page	Exercise 01 Creating a XNA Project	1468494542	view	2	Admin	User	Stephen	Developing Computer Games	5	Creating an XNA Project and Getting Text On Screen	202,205,203, 204,206,208, 201,211,210, 212,209
15389	205	70	6	20	28	url	Introduction to XNA	1468494573	view	2	Admin	User	Stephen	Developing Computer Games	5	Creating an XNA Project and Getting Text On Screen	202,205,203, 204,206,208, 201,211,210, 212,209
15390	201	70	6	13	1	lesson	Exercise 02 XNA Writing to the Screen	1468494586	start	2	Admin	User	Stephen	Developing Computer Games	5	Creating an XNA Project and Getting Text On Screen	202,205,203, 204,206,208, 201,211,210, 212,209
15394	201	70	6	13	1	lesson	Exercise 02 XNA Writing to the Screen	1468494774	start	2	Admin	User	Stephen	Developing Computer Games	5	Creating an XNA Project and Getting Text On Screen	202,205,203, 204,206,208, 201,211,210, 212,209

## 8.3 APPENDIX C –SURVEYS

### 8.3.1 Moodle usage survey

#### Moodle Usage Questionnaire

Thank you for consenting to take part in this survey, it will be contributing to research undertaken in to how lecturers and teachers can be assisted in using VLE's effectively without adding to their workload.

Any information gathered will not be associated with any individual or department, results from the survey will be published in a summary format as part of an Msc by Research awarded by Lincoln University

**Please select up to three of the following that best describes the ways in which you use the Moodle?\***

- ☐ A store of course reference materials
- ☐ A system for communication with students
- ☐ A system for self study for independent learning
- ☐ A system for formative assessment
- ☐ A system for summative assessment
- ☐ A system for group work and collaboration

**When organising materials and activities on Moodle which option best describe the way you group items? \***

- ☐ Adhoc no structure
- ☐ Group by week
- ☐ Group by subject area (e.g. writing skills, presentation skills)
- ☐ Group as per your scheme of work
- ☐ Group by file/activity type e.g. (presentations, notes, worksheets)

**How do you evaluate the success of your Moodle course design?**

- ☐ Student success rates on the course
- ☐ Student grade levels e.g. how many attained P, M or D grades
- ☐ Statistical analysis
- ☐ Survey
- ☐ Informal feedback from students
- ☐ Gut feeling
- ☐ I do not evaluate the course

☐ Other:

**Please identify your level of ability in gathering data about your students activity on a Moodle course. \***

	1	2	3	4	5	
I am unable to access student usage data	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	I am fully able to obtain student access information from the VLE

**Roughly what percentage of your Moodle materials and activities are vital for a student to complete your course (0-100)?**

**Please could you rate how useful the information below is in relation to evaluating the effectiveness of resources/activities you provide via Moodle.**

	Not Useful	Slightly Useful	Useful	Important	Essential
Knowing how many of your student group accessed each resource	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Knowing how many total accesses there were to a Moodle resource (popularity).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Being able to identify resources that had low or zero usage.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Being able to identify resources that were used based on a students level of achievement	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Being able to find an individual students useage	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

### 8.3.2 Moodle Log Analyser Survey

## Moodle Effectiveness Questionnaire

Thank you for consenting to take part in this survey, it will be contributing to research undertaken in to how lecturers and teachers can be assisted in using VLE's effectively without adding to their workload.

Any information gathered will not be associated with any individual or department, results from the survey will be published in a summary format as part of an Msc by Research awarded by Lincoln University

**Please select up to three of the following that best describes the ways in which you use the Moodle?\***

- ☐ A store of course reference materials
- ☐ A system for communication with students
- ☐ A system for self-study for independent learning
- ☐ A system for formative assessment
- ☐ A system for summative assessment
- ☐ A system for group work and collaboration

**When organising materials and activities on Moodle which option best describes the way you group items? \***

- ☐ Adhoc no structure
- ☐ Group by week
- ☐ Group by subject area (e.g. writing skills, presentation skills)
- ☐ Group as per your scheme of work
- ☐ Group by file/activity type e.g. (presentations, notes, worksheets)

**How do you evaluate the success of your Moodle course design?**

- ☐ Student success rates on the course
- ☐ Student grade levels e.g. how many attained P, M or D grades
- ☐ Statistical analysis
- ☐ Survey
- ☐ Informal feedback from students
- ☐ Gut feeling
- ☐ I do not evaluate the course
- ☐ Other:

**Please identify your level of ability in gathering data about your students activity on a Moodle course.\***

	1	2	3	4	5	
I am unable to access student usage data	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	I am fully able to obtain student access information from the VLE

**Roughly what percentage of your Moodle materials and activities are vital for a student to complete your course (0-100)?**

	Not Useful	Slightly Useful	Useful	Important	Essential
Knowing how many of your student group accessed each resource	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Knowing how many total accesses there were to a Moodle resource (popularity).	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Being able to identify resources that had low or zero usage.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Being able to identify resources that were used based on a students level of achievement	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Being able to find an individual students useage	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please could you view the 1-minute video below

The video demonstrates software that represents how students used a Moodle course. [https://www.youtube.com/watch?v=3HZ\\_FnwPP4A](https://www.youtube.com/watch?v=3HZ_FnwPP4A) this link will open a new window, you will need to close this to get back to the form.

**If this software was made available to you to help you decide what resources and activities worked well with your students, would you use it?**

- ☐ Yes  
☐ Yes with training and support  
☐ Maybe  
☐ No I would not  
☐ Other:

**How important is identifying when a student has accessed your course materials and activities held on Moodle for establishing their rate of progress in learning your subject? \***

1 2 3 4 5

Unimportant ☐ ☐ ☐ ☐ ☐ Crucial

**Do you have sufficient time in your role to check Moodle and contact students who have not accessed your Moodle resources and activities? \***

- ☐ Yes
- ☐ No

**If this process was automated for you would you see this as beneficial for you and the student?**

- ☐ Yes for myself and the student
- ☐ Yes for myself
- ☐ Yes for the student
- ☐ No